Generalization of the Shortest Path Approach for Superpixel Segmentation of Omnidirectional Images

Rémi Giraud, Rodrigo Borba Pinheiro, Yannick Berthoumieu

Generalized Shortest Path-based Superpixels for 3D Spherical Image Segmentation

Rémi Giraud[a,*], Rodrigo Borba Pinheiro[b,c] and Yannick Berthoumieu[c]

[a]Bordeaux INP, Univ. Bordeaux, CNRS, IMS UMR 5218, 351 cours de la libération, Talence Cedex, 33400, France
[b]InterDigital, Inc., Le Rainbow, 975 Av. des Champs Blancs, Cesson-Sévigné, 35510, France
[c]Univ. Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des signaux et systèmes, Gif-sur-Yvette, 91190, France

ARTICLE INFO

Keywords:
3D Spherical images
Superpixels
Unsupervised segmentation
Shape regularity

ABSTRACT

The growing use of wide angle image capture devices and need for fast and accurate image analysis in computer visions have enforced the need for dedicated under-representation approaches. Most recent decomposition methods segment an image into a small number of irregular homogeneous regions, called *superpixels*. Nevertheless, these approaches are generally designed to segment standard 2D planar images, *i.e.*, captured with a 90° angle view without distortion. In this work, we introduce a new general superpixel method called SphSPS (for Spherical Shortest Path-based Superpixels)[1], dedicated to wide 360° spherical or omnidirectional images. Our method respects the geometry of the 3D spherical acquisition space and generalizes the notion of shortest path between a pixel and a superpixel center to this context, to fastly extract relevant clustering features. We demonstrate that considering the geometry of the acquisition space to compute the shortest path enables to jointly improve the segmentation performance and the shape regularity of the superpixels. To evaluate this regularity aspect, we also generalize a global regularity metric to the spherical space to address the limitations of the only existing spherical compactness measure. Finally, the proposed SphSPS method is validated on the reference 360° spherical panorama segmentation dataset and on synthetic road omnidirectional images. Our method significantly outperforms both planar and spherical state-of-the-art approaches in terms of segmentation performance and regularity, and we also demonstrate its robustness to noise.

1. Introduction

Many computer vision pipelines now use under-representation or low-level segmentation approaches to overcome the growth in resolution and quantity of image data, which may lead to an important computational load. Among existing approaches, irregular image decomposition techniques were mainly popularized with [1], to decompose an image into *superpixels*, *i.e.*, small, connected regions having homogeneous colors. The image domain is thus generally reduced to hundreds of regions instead of millions of pixels. By processing such regions in an under-representation scale that fit to the image content, the result may be obtained in a very fast manner while being very close to the optimal result at pixel scale.

Over the years, superpixel methods have been successfully applied for many computer vision tasks such as: semantic segmentation [43, 47, 34, 12], object tracking [35, 32, 23], optical flow estimation [28, 48] or style transfer

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

*Corresponding author
remi.giraud@ims-bordeaux.fr (R. Giraud); rodrigo.borbapinheiro@interdigital.com (R. Borba Pinheiro); yannick.berthoumieu@ims-bordeaux.fr (Y. Berthoumieu)
ORCID(s):

[25]. However, with such under-representation, the structural irregularity between adjacent regions may become an issue to use standard neighborhood-based tools. Several works have then proposed different methods to address this setback, *e.g.*, graph-based approach [18], neighborhood structure [15], or hierarchical superpixel decomposition [29].

Additionally to the increase of image resolution, new acquisition devices capturing wide angles, such as fish eyes, or covering a 360^o field of view have become increasingly popular. With these devices, the entire environment can be captured to offer a global understanding of a scene. Hence, in applications such as autonomous driving, spherical or omnidirectional images are particularly interesting and may be used for semantic segmentation [54, 55]. Moreover, with the joint use of a depth-aware system, the captured intensity can be projected on a 3D point cloud, offering additional modalities. Generally, the captured 360^o environment is projected on a discrete 2D plane, generating an equirectangular image. Naturally, such projection on a rectangular space introduces distortions [59]. In this context, standard planar superpixels algorithm have been used on such equirectangular images [8, 37]. Nevertheless, they are not entirely suited since they do not consider the geometry distortions in the projected image, which may limit the segmentation accuracy and the ease of interpretation once visualized in the spherical acquisition space [58].

The majority of existing superpixel methods concern standard 2D planar images. To achieve such irregular decompositions, grouping spatially close pixels with homogeneous colors, many approaches have been proposed using region growing [24], graph-based energy [26], eikonal-based [7], watershed [27], coarse-to-fine [56], or even bayesian algorithms [44]. Some of these methods may offer interesting properties, in terms of exact control of the number of generated superpixels or the ability to set the spatial regularity constraint to produce more or less grid-like decompositions. Among the vast literature of superpixel works, a significant breakthrough was obtained with the SLIC algorithm [1]. This iterative method applies a pixel-wise K-means algorithm to a small area around each superpixel barycenter, which are initialized as a regular grid. Pixels are thus naturally grouped according to a trade-off between distances in the spatial and color spaces (CIELab). The method requires the approximate number of superpixel to generate and a spatial constraint term to produce a decomposition in a reduced processing time. Nevertheless, SLIC may fail to jointly capture object borders and provide regular shapes, while being highly sensitive to texture and noise. Many methods have been proposed based on the SLIC algorithm, introducing boundary constraint [57], advanced feature space [9], non-iterative clustering [2], or a shortest path approach [17].

More recently, deep learning frameworks have also been proposed to decompose an image into superpixels, *e.g.*, [21, 52]. In [21] for instance, the SLIC algorithm is made differentiable to learn adequate image features. Along with the usual deep learning requirements in terms of computational resources and substantial training time and image dataset, these methods may have limited applicability to other images, and even fail to handle large ones due to memory issue. Finally, contrary to most other methods, they may not allow to set the shape regularity which may be an important

property to set according to the application [16]. Hence, unsupervised approaches remain of interest, while they may still consider features extracted from deep learning-based pipelines as input.

For spherical images, the segmentation approach of [13] for 2D planar images has been adapted in [53]. Nevertheless, it uses the same graph-based clustering that generates very irregular regions in terms of shape and size that may not be considered as superpixels. The K-means-based iterative clustering method of SLIC [1] has been extended in [58] to generate spherically regular superpixels, *i.e.*, having more consistent size and shape in the spherical domain. The pixel positions are projected on the unit sphere to compute the spatial constraints to gather homogeneous pixels not spatially too far in the spherical space. With the interest of having visually regular superpixels when working in the spherical domain, the respect of the acquisition space geometry enables a more accurate segmentation of the image objects [58]. Recently, based on the same framework, a non-iterative and a hierarchical extension have been proposed in [11].

These methods use the same features as SLIC, *i.e.*, simple color and spatial distance between the pixel and the average superpixel features. Therefore, as SLIC, they are very sensitive to highly textured areas and presence of noise in the images, which can lead to poor segmentation results. These approaches also do not integrate any contour information, reducing the object segmentation accuracy and leading their algorithm to ignore thin contours. These limitations due to the simple SLIC clustering framework are tackled in [17] for standard 2D planar image segmentation, where the notion of shortest path between the pixel and the superpixel is introduced to extract more relevant features providing higher segmentation accuracy.

Contributions

In this work, we propose an accurate method called SphSPS (Spherical Shortest Path-based Superpixels), to decompose a 2D equirectangular image into 3D spherically regular superpixels. The pipeline of the SphSPS method, considering both 2D and 3D positions is summarized in Figure 1. The core of SphSPS is based on the adaptation of the K-means iterative clustering framework [1] to the spherical space by [58], but addresses its limitations in terms of accuracy, robustness and computational time.

To this end, we generalize of the notion of shortest path between a pixel and a superpixel [17] to our spherical case to extract more relevant features during the clustering process. With such path, when comparing a pixel to a superpixel, the color of the pixels along the path to the superpixel barycenter can be considered to ensure the color homogeneity. Moreover, contour information provided by any prior contour map can also be integrated to prevent the segmentation from crossing object contours in our method. With such features, and a fast implementation of the spherical shortest path, SphSPS is able to converge in very limited processing time and generate accurate and regular spherical superpixels (see Figure 1).

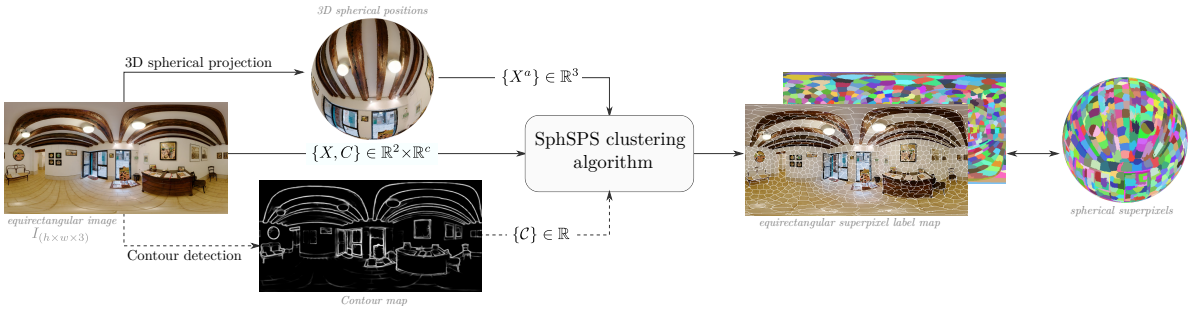


Figure 1: Pipeline of the SphSPS method. A projection in the spherical space is computed from the equirectangular image I to obtain 3D coordinates X^a. Along with 2D positions X and colors C (c channels), a contour map $\mathcal{C}$ can be provided (dotted arrows) to enforce the respect of image contours. SphSPS provides regular superpixels in the spherical space

We also propose a generalization of the global regularity metric [16] to relevantly evaluate the regularity aspect in the spherical space. The performance of SphSPS is compared to the ones of planar and spherical state-of-the-art methods on the reference 360° spherical panorama segmentation dataset (PSD) [45] and synthetic omnidirectional road images from the Omniscape project [39]. SphSPS obtains the highest segmentation and contour detection accuracy while providing regular superpixels in the spherical space, and is robust to noise, contrary to most state-of-the-art methods.

This paper is an extension of the work proposed in [14], with significant improvements and extensive additional experiments. Compared to [14], we provide: (i) more details on our method, with a substantial analysis of its parameters (*e.g.*, sampling frequency of the shortest path, initialization strategies), (ii) a modification of the Generalized Global Regularity (G-GR) metric to more relevantly evaluate the smoothness of the superpixel boundaries, along with a quantitative comparison to the spherical compactness measure proposed in [58]; (iii) an evaluation of the robustness of methods to noise, including other state-of-the-art approaches (ETPS [56], GMMSP [3] and SphLSC [9, 58]) and comparison to an implementation of SphSPS using a linear path; (iv) an additional validation on a synthetic 360° road images [39].

2. Spherical Shortest Path-based Superpixels

In this section, we present the proposed SphSPS method. First, we describe the initial K-means clustering algorithm used in SLIC [1] (Section 2.1). Then, in Section 2.2, we present a brief overview of the spherical geometry system and equirectangular projection (Section 2.2.1) and the adaptation of the K-means clustering algorithm to spherical images introduced in [58], which is the basis of our method (Section 2.2.2). Finally, we present the linear path approach of [17], extracting relevant clustering features on a planar shortest path (Section 2.3.1) and our generalization to the spherical space (Sections 2.3.2 and 2.3.3).

2.1. Planar K-means Iterative Clustering

The proposed SphSPS method is based on the Simple Linear Iterative Clustering (SLIC) [1]. The algorithm is particularly interesting since it only requires as input parameters the approximate number of superpixels to produce and a shape regularity constant, and it may achieve accurate performance in a few iterations. The iterative K-means clustering is spatially constrained through the whole process to respect a spatial regularity in the segmentation. First, superpixels, denoted S_i in the following, are initialized over the image domain as square blocks of size $s\times s$. Each superpixel is described by the average CIELab colors C_{S_i} and spatial barycenter position $X_{S_i} = [x_i, y_i]$ of all pixels within S_i. At each iteration, each superpixel S_i is considered and compared to all pixels $p = [C_p, X_p]$, of color C_p having their position X_p into a $(2s+1)\times(2s+1)$ square window A_i around its barycenter X_{S_i}. At the end of each iteration, a pixel p is associated to the superpixel minimizing the distance D composed of a color d_c and a spatial distance d_s such as:

$$d_c(p, S_i) = \|C_p - C_{S_i}\|_2^2, \tag{1}$$

$$d_s(p, S_i) = \|X_p - X_{S_i}\|_2^2, \tag{2}$$

$$D(p, S_i) = d_c(p, S_i) + d_s(p, S_i)\frac{m^2}{s^2}, \tag{3}$$

where m is the trade-off parameter that enables to set the shape regularity. Finally, a post-processing step is applied to ensure the region connectivity.

2.2. Adaptation to Spherical Images

2.2.1. Spherical Geometry

The correspondence between the planar equirectangular 2D space and the 3D spherical space, can be seen as the respective projection of vertical and horizontal planar coordinates on the meridians and circles of latitude of the sphere, resulting in a spherical image having a width w twice superior to its height h. The projection system is illustrated in Figure 2. Each image pixel $X = [x, y]$ in the 2D planar space, corresponds to a 3D position $X^a = [x^a, y^a, z^a]$ in the spherical acquisition space such as:

$$X = \begin{bmatrix} x = \lfloor \frac{\theta w}{2\pi} \rfloor \\ \\ y = \lfloor \frac{\phi h}{\pi} \rfloor \end{bmatrix} \leftrightarrow X^a = \begin{bmatrix} x^a = \sin(\frac{y\pi}{h})\cos(\frac{2x\pi}{w}) \\ y^a = \sin(\frac{y\pi}{h})\sin(\frac{2x\pi}{w}) \\ z^a = \cos(\frac{y\pi}{h}) \end{bmatrix}. \tag{4}$$

With $\theta = \arctan2(y^a, x^a)$ the azimuthal angle, and $\phi = \arccos(z^a)$ the polar angle. Note that when going from 3D to 2D domain, $x \in [-\frac{w}{2}, \frac{w}{2}]$. Hence, to map x to the image domain, we compute $x \leftarrow x + w$, if $x \leq 0$.

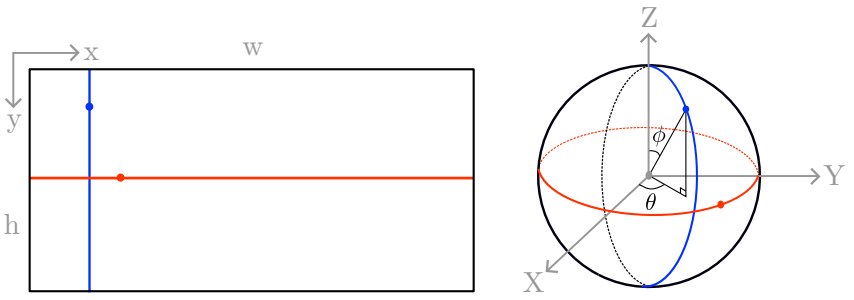


Figure 2: 2D Equirectangular to 3D spherical space projection system. Vertical and horizontal planar coordinates correspond to the meridians and circles of latitude of the sphere. The azimuthal angle θ and polar angle ϕ are considered in the projection system (4)

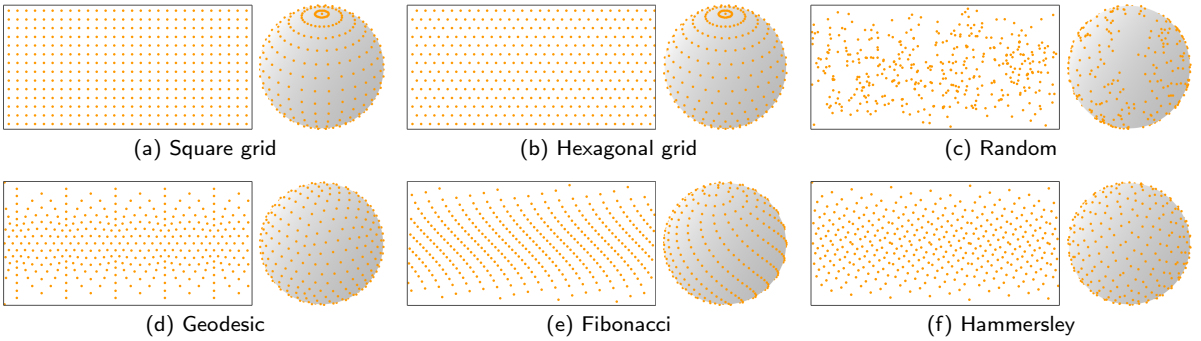


Figure 3: Comparison of superpixel seeds initialization strategies in the planar and spherical space

2.2.2. Spherical K-means Iterative Clustering

SphSPS adapts the planar K-means method to the spherical geometry in three steps, in the same manner as [58]. The first one is the initialization of the K superpixels, to uniformly sample the 3D sphere. In Figure 3, we compare several planar and spherical seed initialization strategies for a number $K = 400$ superpixels. With planar-based sampling (Figures 3(a) and (b)), the seeds are regularly spaced in the two dimensions in the equirectangular image but only along the circles of latitude in the spherical space, so many seeds are set close to the poles. A random seed initialization (Figure 3(c)) may cover the sphere volume but may lead to irregularities, especially with a low number of superpixels. A geodesic approach, relying on subdivisions of icosahedron, can be used to sample the 3D sphere very regularly (Figure 3(d)) but the number of seeds is limited to $K \in \{5\times2^{2n+1} + 2, \text{ with } n \in \mathbb{N}\}$. Finally, two other approaches are compared to uniformly sample the 3D sphere using the Fibonacci sequence [42] (Figure 3(e)) or the Hammersley sampling [49] (Figure 3(f)). Both generate in real-time a sufficiently regular spherical sampling with the desired number of superpixels. In Section 4.2, we compare the use of each initialization strategy on the segmentation performance. According to these results, we use in SphSPS the Hammersley sampling [49], as in [58].

The second step is the adaptation of the definition of the search area, which must consider the position of pixels in the spherical space. For instance, superpixels near the sphere poles must have larger search areas in the equirectangular

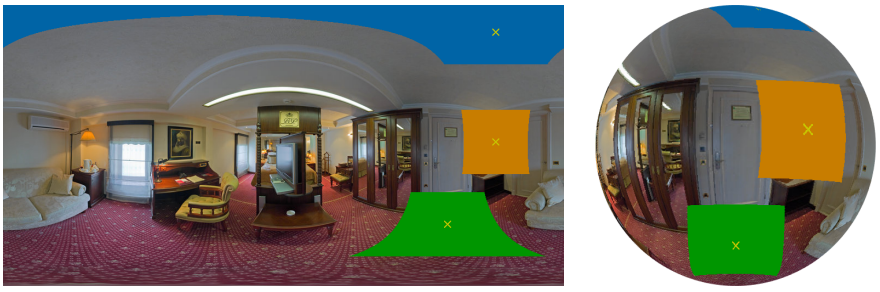


Figure 4: Examples of spherical search areas for different superpixel barycenter positions with $K = 400$ superpixels

image. The overall number of superpixels must also be considered to respect the regularity of the segmentation. Hence, the search area A_i for each superpixel S_i of barycenter $X_{S_i} = [x_i, y_i]$, is defined as:

$$A_i = \{[x, y] | x_i - \frac{S}{\sin(\phi)} \leq x \leq x_i + \frac{S}{\sin(\phi)}, y_i - S \leq y \leq y_i + S\}, \tag{5}$$

where $S = w/\sqrt{K\pi}$ is the average superpixel size and $\phi = y\pi/h$ is the polar angle corresponding to the y-th row for an image of height h and width w. Since we deal with equirectangular images, the 360° aspect must also be handled to connect the pixels at the horizontal boundaries. This is simply done using a horizontal warping of positions, when the search region falls outside the image boundaries [58]. The spherical search areas are represented for several superpixel barycenter positions in Figure 4.

The third modification to implement in the planar algorithm is the computation of the spatial distance d_s (3) in the spherical space. To compute a spatial distance in this 3D space, *i.e.* using 3D positions X^a (4), we can use the Euclidean one, defined such as: $d_s(X^a_p, X^a_{S_i}) = \|X^a_p - X^a_{S_i}\|_2^2$. Nevertheless, as proposed in [58], we choose to use in SphSPS a spherical and computationally costless cosine dissimilarity distance defined as:

$$d_s(X^a_p, X^a_{S_i}) = 1 - \left\langle X^a_p, X^a_{S_i} \right\rangle, \tag{6}$$

with $\left\langle X^a_p, X^a_{S_i} \right\rangle$, the scalar product between the two position vectors. Note that, although it does not exactly follow the sphere geometry, the Euclidean and Cosine dissimilarity distances can achieve almost similar performance for [58] with adjusted parameter m (3) (see Section 4).

2.3. Generalized Shortest Path Method

2.3.1. Feature extraction on a shortest path

In [17], when processing a pixel p, the color and contour information of the pixels q on the planar shortest path $\mathbf{P}_{p,S_i}$ towards the barycenter of the superpixel S_i are considered in the clustering distance. By using these additional features, the clustering accuracy and the respect of object contours are improved. Also, contrary to color-based geodesic paths

used for instance in [46, 36], the planar shortest path increases the shape regularity which is a desirable property [16], by enforcing homogeneous star-convex shapes with respect to their barycenters [20]. SphSPS integrates these features using the same clustering distance D as [17], but with a different shortest path definition since it is defined according to the spherical space.

First, the color information along the path is compared to the average one of the superpixel to ensure the color homogeneity. This way, the regularity is increased in a relevant manner, preventing non convex shapes to appear [17]. The color distance of the pixels q on the path $\mathbf{P}_{p,S_i}$ is added to the color distance d_c (1) such that:

$$d_c(p, S_i, \mathbf{P}_{p,S_i}) = \lambda d_c(p, S_i) + \frac{1-\lambda}{|\mathbf{P}_{p,S_i}|} \sum_{q \in \mathbf{P}_{p,S_i}} d_c(q, S_i), \tag{7}$$

where λ is a trade-off parameter between the color of the considered pixel and the ones on the shortest path, which is usually set to 0.5.

Along the shortest path, a contour information can also be extracted to ensure the respect of objects borders. These borders may indeed not be in line with the ones obtained following the color homogeneity criteria. Especially for thin contours, which are not yet captured in the clustering framework. Any contour map, denoted C, with normalized values between 0 and 1, respectively indicating the absence or presence of contours, can be computed and used to define a contour term d_C such that:

$$d_C(\mathbf{P}_{p,S_i}) = 1 + \gamma \max_{q \in \mathbf{P}_{p,S_i}} C(q), \tag{8}$$

where $\gamma \geq 0$ is the parameter penalizing the crossing of a contour. With such term, the pixel p most likely will not be associated to the superpixel when a high contour intensity is found on the path.

Finally, the clustering distance D of the SphSPS method ensuring more regular regions and respecting the image object contours, is defined as:

$$D(p, S_i) = \left(d_c(p, S_i, \mathbf{P}_{p,S_i}) + d_s(p, S_i) \frac{m^2}{s^2} \right) d_C(\mathbf{P}_{p,S_i}), \tag{9}$$

where d_s is the spherical spatial distance using the cosine dissimilarity (6), and $\mathbf{P}_{p,S_i}$ is the proposed spherical shortest path computed as follows.

2.3.2. Generalized shortest path

For any image to segment, the shortest path should be computed in the acquisition space, to respect the acquisition geometry. In Figure 5, we illustrate several examples of shortest paths in the planar space as in [17], and in the spherical

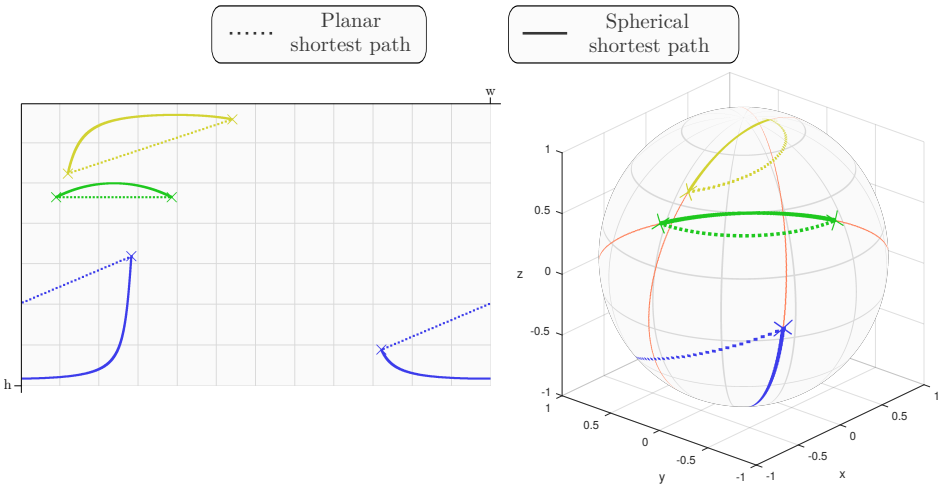
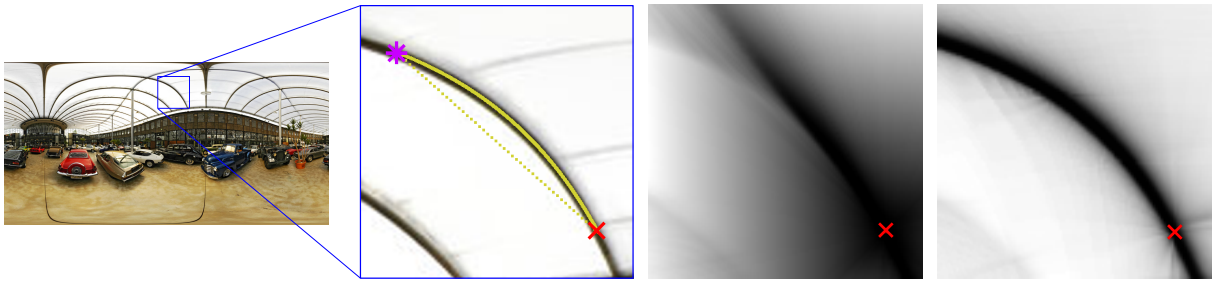


Figure 5: Examples of planar (dotted lines) and spherical shortest paths (full lines) between points in the 2D planar image space (left) and 3D acquisition space (right). The spherical path follows the shortest geodesic path on the sphere



(a) Equirectangular image (b) Linear shortest path distance (c) Spherical shortest path distance

Figure 6: Comparison of linear shortest path (dotted line) and spherical shortest path (full line) between an initial (red cross) and final (purple star) pixel position in (a). In (b) and (c), the average color distance on all points of the path from the the initial point (star) is represented (lighter color indicates a higher distance). When trying to associate a point to a superpixel, such distance from the point to the superpixel barycenter is considered (7). Hence, using the spherical shortest path enables to follow the structure of objects more accurately in the equirectangular image

one as in SphSPS. The difference of using these paths to compute the color distance on the path to the superpixel barycenter (7) is also illustrated in Figure 6. For a reference point (red cross), the average color distance to all the other points on linear and spherical shortest paths are respectively represented in Figure 6(b) and 6(c). The spherical path enables the algorithm to respect, within the equirectangular image, the linear structure of objects in the spherical space.

For the planar case, the acquisition and image spaces are equivalent. Hence, the shortest path can be reduced to a linear path that can be easily computed with a discrete algorithm such as [6]. Nevertheless, in the general case with spherical or even circular devices using fisheyes with different capture angles, distorsions can be introduced between the acquisition and the image space ($\mathbb{N}^2$). Therefore, the shortest path should be first explicitly computed in the acquisition space, denoted as $\mathbf{P}^a_{p,S_i}$, and projected back to the planar space to provide the discrete sample positions in the 2D image.

Hence, the general formulation of the discrete shortest path is defined as:

$$\mathbf{P}_{p,S_i} = \mathbf{P}^a_{p,S_i} \quad \overrightarrow{\text{proj}} \quad \{\mathbb{N}^2\}. \tag{10}$$

2.3.3. Shortest path in the spherical space

The shortest path in the spherical space consists in following the geodesic along the sphere [19]. This geodesic lies within the *great circle* (in orange color in Figures 5 and 7), containing the two considered points and the center of the sphere to form a disc. In the following, we rely on the Slerp (Spherical linear interpolation) formulation of the spherical geodesic path problem proposed in [40] to sample the points of our shortest path. Note that tangential approaches to extract way-points on the great circle have been formalized, for instance in [22], but such theoretical methods use many trigonometric computations that may impact the processing time. To the best of our knowledge, no direct algorithm, such as [6] for the planar case, has been made explicitly available to extract in an equirectangular image the discrete shortest path between two pixels in the spherical space.

Spherical geodesic path implementation

The proposed process to compute the spherical shortest path on a great circle is illustrated in Figure 7. For each comparison of a pixel at position X^a_p to a superpixel of spatial barycenter $X^a_{S_i}$, we first compute an orthogonal coordinate system $[\vec{X}^a_p, \vec{X}^{a*}_{S_i}]$ that lies within their great circle. To build such system, we use the Gram-Schmidt orthogonalization process [5] to get the position $X^{a*}_{S_i}$, an orthogonal vector to X^a_p within the great circle such as:

$$X^{a\ *}_{S_i} = \frac{X^a_{S_i} - \left\langle X^a_p, X^a_{S_i} \right\rangle X^a_p}{\left\| X^a_{S_i} - \left\langle X^a_p, X^a_{S_i} \right\rangle X^a_p \right\|_2}. \tag{11}$$

The cost of computing (11) is greatly reduced since the scalar product $\left\langle X^a_p, X^a_{S_i} \right\rangle$ has already been computed for the spatial distance d_s (6). Then, the angle α between the two points is simply obtained with $\alpha = \arccos\left(\left\langle X^a_p, X^a_{S_i} \right\rangle\right)$. Finally, α is used to linearly sample the geodesic path $\mathbf{P}^a_{p,S_i}$, starting from the pixel position and progressively reaching the superpixel barycenter such as:

$$\mathbf{P}^a_{p,S_i} = \cos(\alpha_\mathbf{N}) X^a_p + \sin(\alpha_\mathbf{N}) X^{a*}_{S_i}, \tag{12}$$

with $\alpha_\mathbf{N} = \frac{[\![0,N-1]\!]}{N-1}\alpha \in \mathbb{R}^N$, intermediate angles to linearly sample N points within the coordinate system $[\vec{X}^a_p, \vec{X}^{a*}_{S_i}]$ between the pixel and barycenter positions. The geodesic path in the acquisition space, is finally projected to the planar

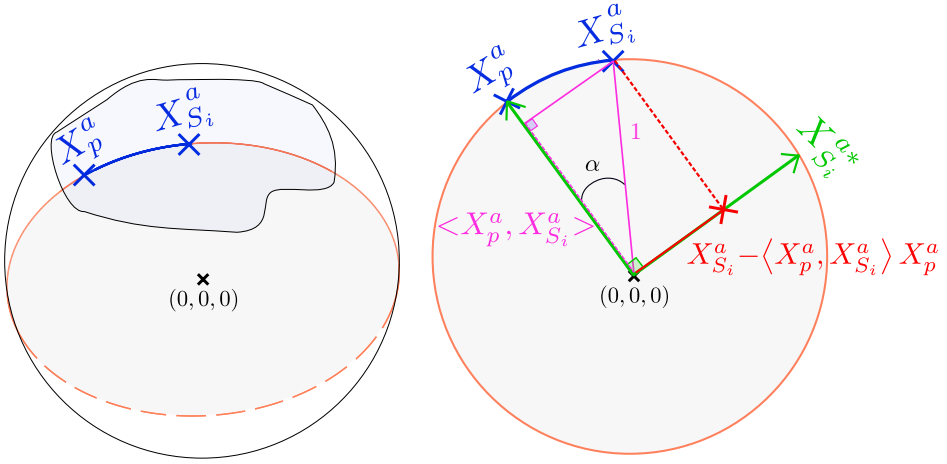


Figure 7: Computation of the spherical shortest path. The orthogonal coordinate system $[\vec{X}^a_p, \vec{X}^{a*}_{S_i}]$ is computed from projection of $X^a_{S_i}$ on X^a_p (11). The angle α between the positions is then used to sample 3D points on the path (12)

space (4) to get the image pixel positions contained into the path $\mathbf{P}_{p,S_i}$ (10). By this way, we compute the discrete shortest spherical path with simple calculations, dividing the processing time by a factor 2 compared to tangential approaches.

Fast computation using recursive implementation

Compared to the complexity of the SLIC method [1], a much larger number of pixels are considered in a shortest path, leading to a significant increase of calculations. Nevertheless, we can reduce the cost of this additional complexity by using a recursive implementation.

First, as in [17], for each superpixel, we store the color distance d_c to each tested pixel, making it directly available for the next pixels to process. This first optimization reduces the processing time by 50%. Here, we propose a general optimization framework using the path redundancy with recursive implementation. The principle is that 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. In the planar case using the algorithm of [6] this assumption is not exactly accurate, but it becomes true in the spherical case since all points lie on the same great circle. Using recursive implementation, we can store the average color distance and maximum contour intensity on the path for each crossed pixel. By this way, within a superpixel search area, for most pixels, the large quantity of information contained in the shortest path has already been computed once and is directly available. This recursive scheme also reduces the processing time by a factor 2.

3. Generalized Global Regularity Measure

Most superpixel methods try to decompose the image into regions having approximately the same size, and being homogeneous in terms of color, so they should naturally be contained into the image objects. This is usually done by

optimizing a trade-off between color and spatial terms. Giving more weight to color would generally provide more accurate superpixels in terms of respect of object contours, while increasing regularity would prevent superpixel borders from following them. Works such as [16] have demonstrated that having regular superpixels may increase the performances of superpixel-based pipelines. Hence, algorithms should try to produce both accurate and regular superpixels and dedicated metrics should jointly evaluate object segmentation and shape regularity performances.

Nevertheless, this last aspect is rarely evaluated in the literature. The standard compactness metric [38] is the only one extended to the spherical space [58], although it was proven very limited [16]. In this section, we propose a more relevant metric to evaluate the shape regularity in any acquisition space.

3.1. Limitation of the Compactness Measure

The first reference regularity measure in the superpixel literature is the compactness COM [38] that evaluates the average circularity of each superpixel shape. In [58], the extension of this metric to the spherical case is proposed. The COM measure independently evaluates each superpixel S_i of a segmentation $\mathcal{S} = \{S_i\}$ such as:

$$\mathrm{COM} = \frac{1}{\sum\limits_{S_i \in \mathcal{S}} |S_i|} \sum_{S_i \in \mathcal{S}} Q_{\mathrm{sph}}(S_i)|S_i|, \tag{13}$$

with Q_{sph} the spherical isoperimetric quotient [33] defined as:

$$Q_{\mathrm{sph}}(S_i) = \frac{4\pi|S_i| - |S_i|^2}{|P(S_i)|^2}, \tag{14}$$

with $P(S_i)$ the superpixel perimeter.

Although this metric has been regularly used when evaluating the regularity, it has been proven very limited [16]. It relies on a reduced circularity criteria such that ellipses can obtain higher regularity measures than squares, and it is highly sensitive to boundary noise and inconsistent with the superpixel size. The same limitations appear for the extension to the spherical case, to such extent that the COM measure may even fail to differentiate spherical and planar-based methods [58].

3.2. Generalized Global Regularity Metric

3.2.1. Global regularity metric

To address the issues of the COM measure, a global regularity metric (GR) has been introduced in [16], containing two terms. First, instead of comparing the superpixels to a circle, it robustly evaluates the convexity, the contour smoothness, and the 2D balanced repartition of each superpixel with the Shape Regularity Criteria (SRC) term. The convexity and smoothness properties are computed with respect to the discrete convex hull corresponding to the

superpixel shape:

$$\mathrm{SRC}(S_i) = \frac{\mathrm{CC}(H_{S_i})}{\mathrm{CC}(S_i)} \mathrm{V}_{\mathrm{xy}}(S_i), \tag{15}$$

where $\mathrm{V}_{\mathrm{xy}}(S_i) = \min(\sigma_x, \sigma_y)/\max(\sigma_x, \sigma_y)$, evaluates the balanced repartition of the shape with σ_x and σ_y the square root of standard deviations of pixel positions x and y in S_i, H_{S_i} is the convex hull containing S_i, and CC measures the ratio between the perimeter and the area of the shape.

As for the compactness COM (13), SRC is independently computed for each superpixel. Hence, [16] introduces a Smooth Matching Factor (SMF) term to also evaluate the consistency of all superpixel shapes. First, each superpixel S_i is registered on its barycenter to obtain S_i^*. Then, these registered superpixel shapes are averaged to compute S^*, the average superpixel shape, created from the superposition of all superpixels. Finally, each registered superpixel S_i^* is compared to the average shape S^* such that:

$$\mathrm{SMF}(S_i) = 1 - \left\| \frac{S^*}{|S^*|} - \frac{S_i^*}{|S_i^*|} \right\|_1 \Big/ 2. \tag{16}$$

Finally, the global regularity is defined with GR combining these two metrics such that:

$$\mathrm{GR}(\mathcal{S}) = \frac{1}{\sum_{S_i \in \mathcal{S}} |S_i|} \sum_{S_i \in \mathcal{S}} |S_i| \mathrm{SRC}(S_i) \mathrm{SMF}(S_i). \tag{17}$$

3.2.2. Generalization in the acquisition space

Ideally, the regularity of each discrete superpixel shape S_i, should be evaluated in the acquisition space. In our context, we first deal with a set of 3D points S_i^a, from the projections of the pixel positions of S_i (4). Since the GR metric relies on the computation of a convex hull for SRC (15), and uses barycenter registration for SMF (16), it cannot be directly applied to such point clouds in $\mathbb{R}^3$.

To be able to use the GR metric, we propose to reduce to 2D shapes relevantly approximating each set of 3D points of S_i^a. The whole projection process is illustrated in Figure 8. Firstly, a superpixel S_i in the discrete image space is projected to its acquisition one, giving a spherical point cloud S_i^a. Secondly, to reduce to a 2D point cloud $S_i^{a \to \{\mathbb{R}^2\}}$, we apply a principal component analysis (PCA) on S_i^a, and project the points on its two most significant eigenvectors. Finally, to avoid holes in the projection and get a dense discrete shape, a downsampling is performed to obtain a discrete 2D shape $S_i^{a \to \{\mathbb{N}^2\}}$, that is finally filled according to its concave hull. Note that in [14], only a downsampling was used, with a higher scale, resulting in a smaller and smoothed 2D shape, thus reducing the relevance of the projection and the ability to measure the the contour irregularities.

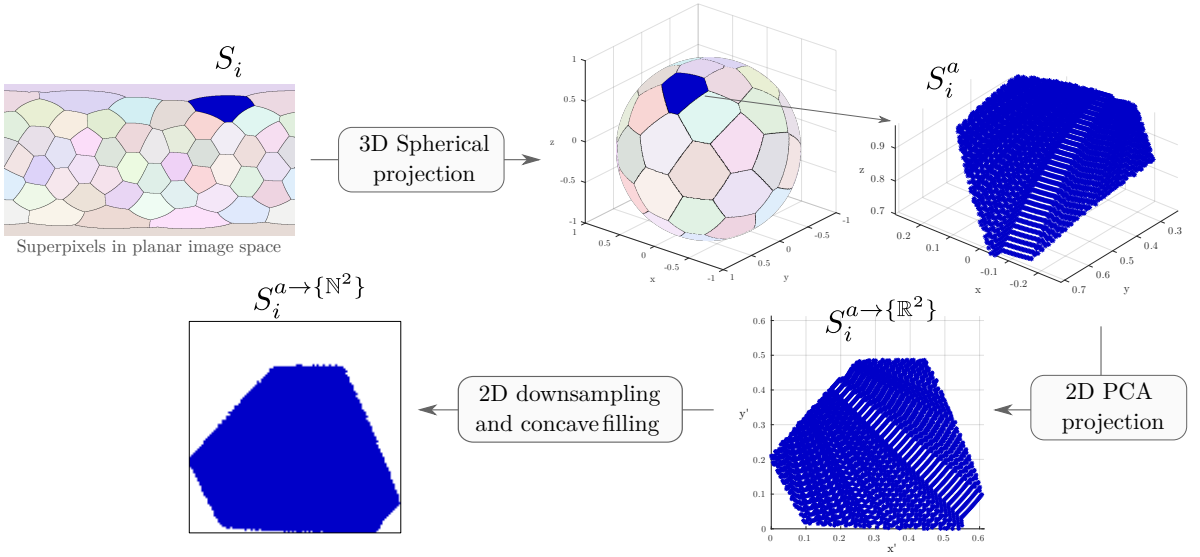


Figure 8: Projection process for the proposed Generalized Global Regularity (G-GR) metric (18). Each superpixel shape S_i from the planar segmentation is projected in its acquisition space to obtain S_i^a. Then, a principal component analysis (PCA) is performed to project the 3D point cloud on a two dimensional space using ($S_i^{a\to\{\mathbb{R}^2\}}$). Finally, it is downsampled to generate a 2D matrix, and the remaining potential holes are filled by considering the concave hull of the shape ($S_i^{a\to\{\mathbb{N}^2\}}$). The G-GR metric especially relying on a convex hull can then be applied on each obtained dense 2D shape

This way, each superpixel shape is relevantly projected from the 3D acquisition space into a discrete 2D shape. The initial SRC and SMF measures can then be used in the proposed Generalized Global Regularity (G-GR) metric as:

$$\text{G-GR}(\mathcal{S}) = \frac{1}{\sum_{S_i\in\mathcal{S}} \left|S_i^{a\to\{\mathbb{N}^2\}}\right|} \sum_{S_i\in\mathcal{S}} \left|S_i^{a\to\{\mathbb{N}^2\}}\right| \text{SRC}(S_i^{a\to\{\mathbb{N}^2\}})\text{SMF}(S_i^{a\to\{\mathbb{N}^2\}}). \tag{18}$$

Regularity results for SphSPS and state-of-the-art methods are compared in Section 4.3. With the proposed G-GR metric, contrary to the COM measure, a performance gap is now visible such that no planar methods have higher regularity than spherical ones for a given number of superpixels.

4. Results

4.1. Validation Framework

4.1.1. Dataset

To validate our method, we considered images from the reference 360º equirectangular dataset SUN360 [50]. In [45], 75 images of 512×1024 pixels from SUN360 are selected and accurately manually segmented to provide a Panorama Segmentation Dataset (PSD). These 75 ground-truth segmentations contain between 115 and 1085 objects having an average size of 1334 pixels. Examples of PSD images are given in Figure 9.

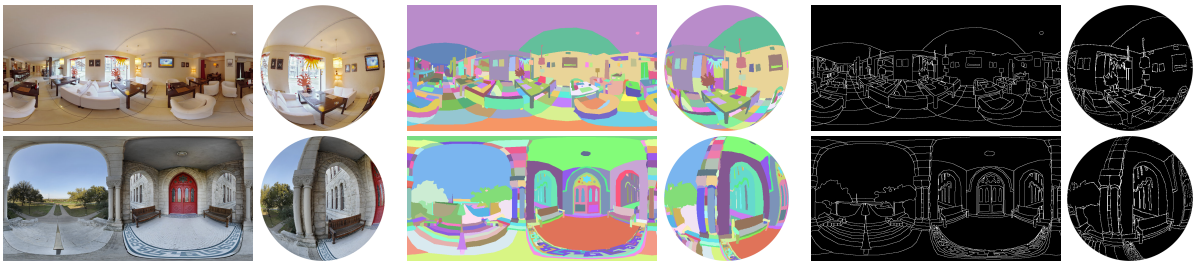


Figure 9: Examples of images from the Panorama Segmentation Dataset (PSD) [45]

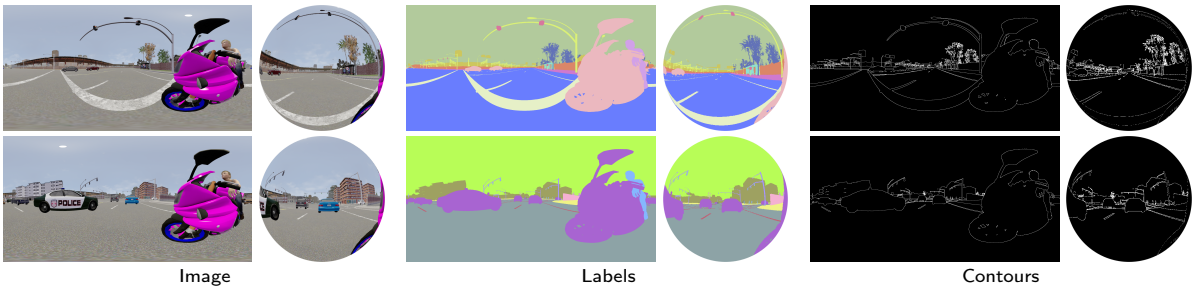


Figure 10: Examples of images from the Omniscape project [39]

To further demonstrate the applicability of the SphSPS method, we also consider a dataset of 100 synthetic omnidirectional road images, provided in the frame of the Omniscape project [39]. Examples of these images, of size 900×1800 pixels and containing up to 13 classes are given in Figure 10.

4.1.2. Metrics

To compare the performance of the SphSPS method to the ones of the state-of-the-art, we use the contour detection, object segmentation and regularity metrics recommended in [16].

In terms of contour detection, the Boundary-Recall (BR) is a commonly employed metric evaluating the compliance of the superpixel boundaries $\mathcal{B}(\mathcal{S})$ to the ground truth contours $\mathcal{B}(\mathcal{G})$ such that:

$$\mathrm{BR}(\mathcal{S},\mathcal{G}) = \frac{1}{|\mathcal{B}(\mathcal{G})|} \sum_{p\in\mathcal{B}(\mathcal{G})} \delta[\min_{q\in\mathcal{B}(\mathcal{S})} \|p-q\| < \epsilon], \tag{19}$$

with ϵ a distance threshold set to 2 pixels [16], and $\delta[a] = 1$ when a is true and 0 otherwise. With BR, each ground truth pixel is considered as detected, if a superpixel boundary is at less than an ϵ distance. The issue of the BR measure, is that it does not take into account the number of superpixel borders, so fuzzy methods can have high performances [16]. Hence, BR results can be compared to the number of pixels of superpixel borders, called Contour Density (CD).

To go further in the evaluation of contour detection performance, the standard Precision-Recall curves can also be represented. These are computed from a contour probability map $\in [0, 1]$ that is generated by averaging the superpixel

borders obtained at different segmentation scales $K \in [50, 3000]$. This normalized boundary map is then thresholded by several intensities $\in [0, 1]$ to get a binary contour map, and for each threshold, the Precision (PR) (percentage of accurate detection among the superpixel borders) is computed along with the BR measure. The performance of each PR/BR curve can be summarized with the maximum standard F-measure defined as:

$$\mathrm{F} = \frac{2.\mathrm{PR}.\mathrm{BR}}{\mathrm{PR} + \mathrm{BR}}. \tag{20}$$

Since the dataset provides segmented regions, object segmentation performance can also be evaluated. To do so, we use the standard Achievable Segmentation Accuracy (ASA) measure [26], which has been proven highly correlated to the Undersegmentation Error [30] in [16]. The ASA evaluates the overlap of the superpixel segmentation $\mathcal{S}$ with the objects of the ground truth segmentation, denoted $\mathcal{G}$, such as:

$$\mathrm{ASA}(\mathcal{S}, \mathcal{G}) = \frac{1}{\sum\limits_{S_i \in \mathcal{S}} |S_i|} \sum_{S_i} \max_{G_j \in \mathcal{G}} |S_i \cap G_j|. \tag{21}$$

Finally, the regularity aspect is robustly evaluated in the acquisition space with the proposed G-GR metric (18). To demonstrate its relevance, we also compare its results to the ones of the spherical COM measure proposed in [58] in Annex A.

4.1.3. Parameter settings

The SphSPS method was implemented in MATLAB using C-MEX code, on a Linux computer having 12 cores at 2.6 GHz with 64GB of RAM. SphSPS is based on the spherical K-means framework of [58], with 5 iterations, with a seed initialization following a Hammersley sampling [49], and a cosine dissimilarity spatial distance d_s (6). It uses the 6 dimension CIELab color space of [9], and includes the features of neighboring pixels [17] in its color distance d_c. In its spherical shortest path, by default $N = 15$ pixels are considered (12). The color distance trade-off parameter λ (7), is set to 0.5 as in [17], and when used, *i.e.*, $\gamma \neq 0$, the contour prior is computed from [51] and γ is set to 10 (8). Finally, the regularity parameter m (9) is empirically set to 0.12 to provide a satisfying trade-off between the spatial regularity of superpixels and their segmentation performance.

4.2. Impact of Contributions

In this section, we show the impact of the contributions introduced in the SphSPS method on contour detection performance using PR/BR curves with maximum F-measure (20), on segmentation accuracy using ASA metric (21), and on regularity with the proposed G-GR metric (18). First, in Figure 11, we quantitatively evaluate the impact of the initialization strategies represented in Figure 3. The spherical seed samplings logically provide much higher

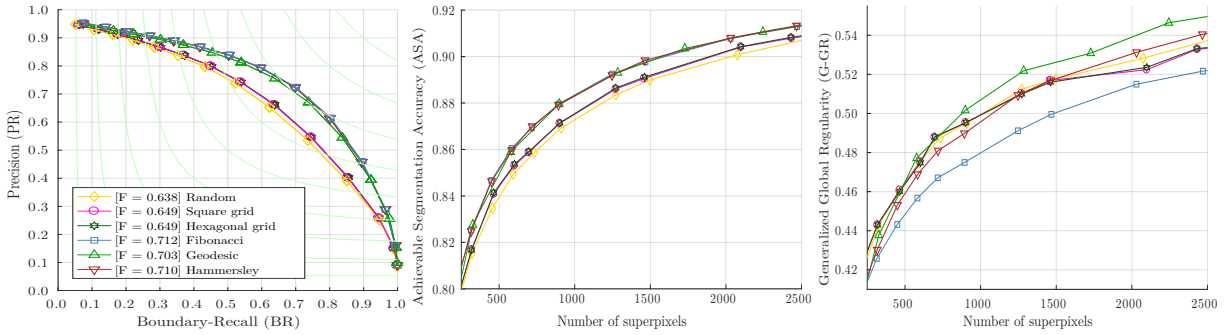


Figure 11: Impact of the seed initialization strategy on the performance of SphSPS evaluated on the PSD dataset [45]. The different sampling strategies are represented in Figure 3

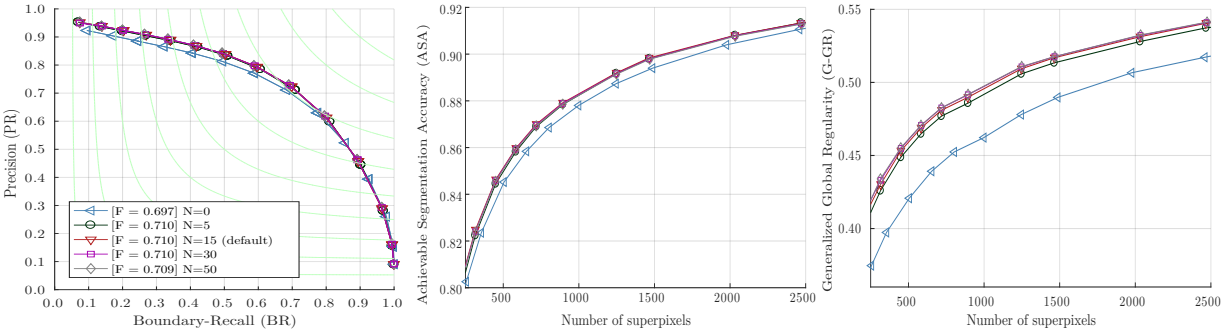


Figure 12: Impact of the number of points N in the spherical shortest path (12) on the performance of SphSPS evaluated on the PSD dataset [45]

accuracy, since superpixels are regularly placed over the acquisition space and can accurately follow object contours. For instance, for a low number of superpixels, planar initialization would set a large number of seeds on top and bottom of the image where very few objects generally are. After a few iterations, we observe that the spherical distance of the SphSPS method is able to provide a spherically regular decomposition with any input method. Nevertheless, with a high number of superpixels, the two most regular spherical initialization methods, Hammersley and Geodesic clearly obtain the highest regularity. In the following, due to the complexity and lack of exact control of number of superpixels with the geodesic approach, we use the Hammersley sampling in the SphSPS method.

Then, in Figure 12, we report the influence of the number of points in the shortest path (12). More sampling points enable to more accurately follow the spherical shortest path to capture homogeneous colors, so the shape regularity is enforced with the number of points, and reaches a plateau around $N = 15$. The same behavior can be observed for the PR curves where a plateau is obtained with a low number of points thanks to our recursive implementation (see Section 2.3.2). In the following we use a trade-off number of $N = 15$ points in the shortest path.

Finally, we show the impact of the different distance settings in SphSPS on PR/BR, ASA and G-GR curves in Figure 13, and a zoom on a segmentation example in Figure 14. With a 3 feature dimension space (3-Lab, λ=1, and γ=0), SphSPS reduces to an implementation of the spherical SLIC algorithm [58]. With the 6 dimension space (6-Lab)

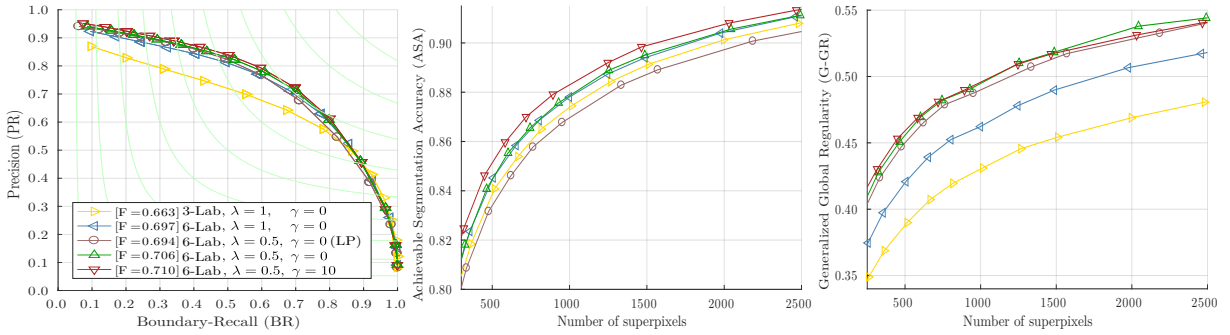
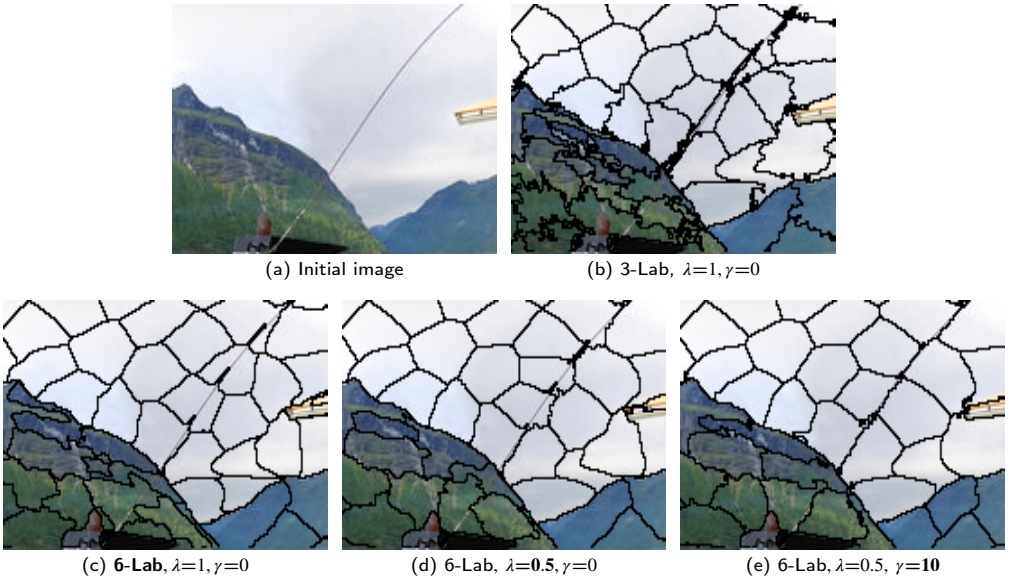


Figure 13: Impact of the distance parameters on the performance of SphSPS evaluated on the PSD dataset [45]. The contributions enable to significantly improve the accuracy and regularity performances



(a) Initial image (b) 3-Lab, λ=1, γ=0

(c) **6-Lab**, λ=1, γ=0 (d) 6-Lab, λ=**0.5**, γ=0 (e) 6-Lab, λ=0.5, γ=**10**

Figure 14: Visual impact of SphSPS parameters. Each contribution relevantly increases the regularity and $\gamma = 10$ integrates the contour prior information

SphSPS uses the CIELab color features of [9] with the information of the neighboring pixels as in [17]. With λ set to 0.5, SphSPS considers in its model the average color distance on the path (7), and with $\gamma = 10$, the maximum contour information (8). We also compare the results obtained with the SphSPS method using a shortest linear path (LP), represented in Figure 5.

We observe that each contribution within the SphSPS method improves the segmentation performance. In particular, we see that the color distance on the shortest path, strengthens the superpixel convexity and homogeneity, to provide much more regular superpixels in a relevant manner that also increases the segmentation accuracy. Finally, these results demonstrate the interest of using our spherical path, since the exact same method using the initial 2D planar shortest path [17] reports lower performance.

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

The performance of the proposed SphSPS method are compared to the ones of state-of-the-art approaches on the standard PSD dataset [45] and also on a set of synthetic images extracted from the Omniscape project [39] on the metrics presented in Section 4.1.2. We consider the recent planar methods SLIC [1], ETPS [56], LSC [9], SNIC [2], SCALP [17], and GMMSP [3]. We also compare our SphSPS to the spherical approach SphSLIC [58], in its two different settings, *i.e.*, using an Euclidean spatial distance (SphSLIC-Euc) and the same Cosine (SphSLIC-Cos) distance as SphSPS (6). For further comparison to spherical methods, we also implement the natural extension of the LSC method [9] to spherical images (SphLSC), *i.e.*, using the SphSLIC framework [58] and the 6 dimension features of [9]. To ensure fair comparison between methods, results are reported using the default regularity setting recommended by the authors, providing a good trade-off between accuracy and regularity. Note that only for the SphSLIC-Cos method [58], no default parameter is provided so we use the one optimizing the segmentation accuracy.

We report the quantitative performance for several numbers of superpixels of SphSPS and the compared state-of-the-art methods in Figures 15 and 16. First, on the PSD dataset (Figure 15), we observe that SphSPS obtains the best segmentation and contour detection performance with the highest ASA (21) and the highest F-measure (20) of $\mathrm{F} = 0.710$, even without using a contour prior ($\gamma = 0$), *i.e.*, only using color information on the shortest path ($\mathrm{F} = 0.706$). A significant improvement is obtained over the other spherical methods SphLSC and SphSLIC using both distances, while producing more regular superpixels. SphSPS indeed succeeds in producing both accurate and spherically regular superpixels. On the synthetic road images (Figure 16), SphSPS also offers the best trade-off on all metrics. While SphSPS is ranked 3rd of F-measure, the two better methods on this criteria (ETPS [56] and SNIC [2]) poorly segment the image objects according to the ASA metric. On this last criteria, SphSPS is ranked 3rd, after the GMMSP method [3] that is the less regular among all methods, and the SphLSC method ([9, 58]) which also produces very fuzzy superpixels and has a low F-measure.

The proposed G-GR (18) provides a relevant regularity measure able to differentiate planar and spherical methods. In particular, with a representative number of superpixels, no planar methods have higher regularity than spherical ones for a given number of superpixels, contrary to the results of the COM metric [58] reported in Annex A. G-GR results are also better correlated to the evolution of the superpixel scale, which is relevant since, to a certain extent, smaller superpixels are more likely to have regular shapes.

Examples of segmentation on the 360^{o} equirectangular images, and projected on the unit sphere, with superpixel borders and labels, for the proposed SphSPS and the best state-of-the-art methods are represented in Figures 17 and 18 for the PSD dataset, and in Figure 19 for the Omniscape images. On both image type, we observe that SphSPS generates very regular superpixels in the spherical space while it is able to accurately capture the image objects compared to other methods.

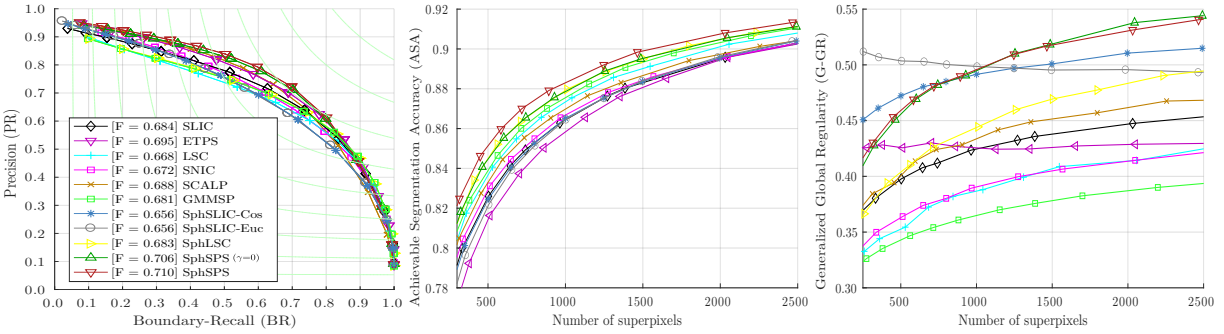


Figure 15: Quantitative comparison on the PSD images [45], on PR/BR, ASA and G-GR metrics of the proposed SphSPS method to the state-of-the-art ones

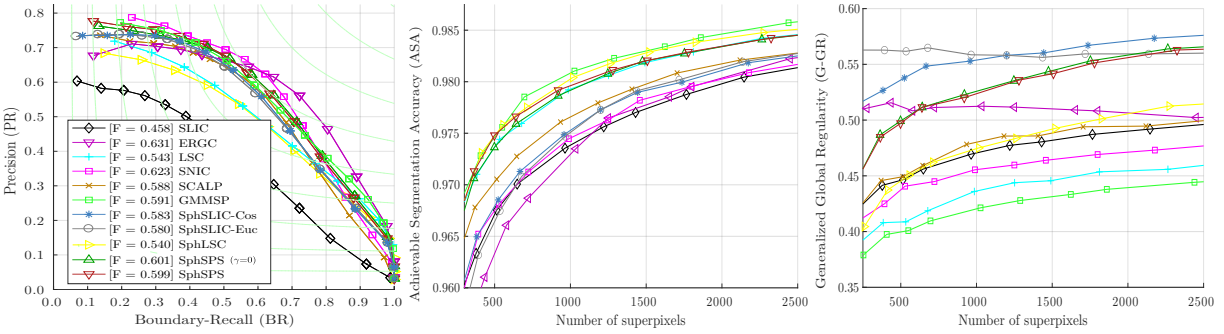


Figure 16: Quantitative comparison on the synthetic Omniscape images [39] of the proposed SphSPS method to the state-of-the-art ones

4.3.1. Robustness to noise

To demonstrate the robustness to noise, we also report the results obtained on PSD images affected by the addition of a white Gaussian noise of variance 20. Performance in terms of PR/BR, ASA and G-GR are reported in Figure 20 with segmentation examples given in Figure 21.

The proposed SphSPS method appears to be significantly robust to noise compared to most other methods that may suffer a dramatic loss of accuracy when applied to noisy images. Visually, the superpixels generated by SphSPS remain regular while other methods provide very fuzzy superpixels, or may even severely fail to capture the image objects. Finally, in Table 1, we report the quantitative results for a given number of $K = 1500$ superpixels for both types (initial and noisy PSD images). In this Table, we also report Boundary Recall performance (BR) (19) with respect to Contour Adherence (CD), *i.e.*, the percentage of superpixel border among image pixels, to further express the fuzzy behaviour of several state-of-the-art methods.

4.3.2. Processing time

The computational cost induced from the larger 6 dimensional feature space [9] and the extraction of color and contour information on the shortest path, is compensated by the fast convergence in a low number of iterations of

Figure 17: Visual comparison between SphSPS and the best planar and spherical (underlined) state-of-the-art methods on PSD images, for two superpixel numbers $K = 1200$ (top-left) and $K = 400$ (bottom right). SphSPS produces regular spherical superpixels with smooth boundaries that adhere well to the image contours

Figure 18: Visual comparison between SphSPS and the best planar and spherical (underlined) state-of-the-art methods on PSD images, for two superpixel numbers $K = 1200$ (top-left) and $K = 400$ (bottom right). SphSPS produces regular spherical superpixels with smooth boundaries that adhere well to the image contours

SphSPS. Without using the shortest path, SphSPS generates superpixels in 0.85s per image of size 512×1024 pixels and already obtains higher accuracy (F = 0.697) than the state-of-the-art methods (see Figure 13). With the information

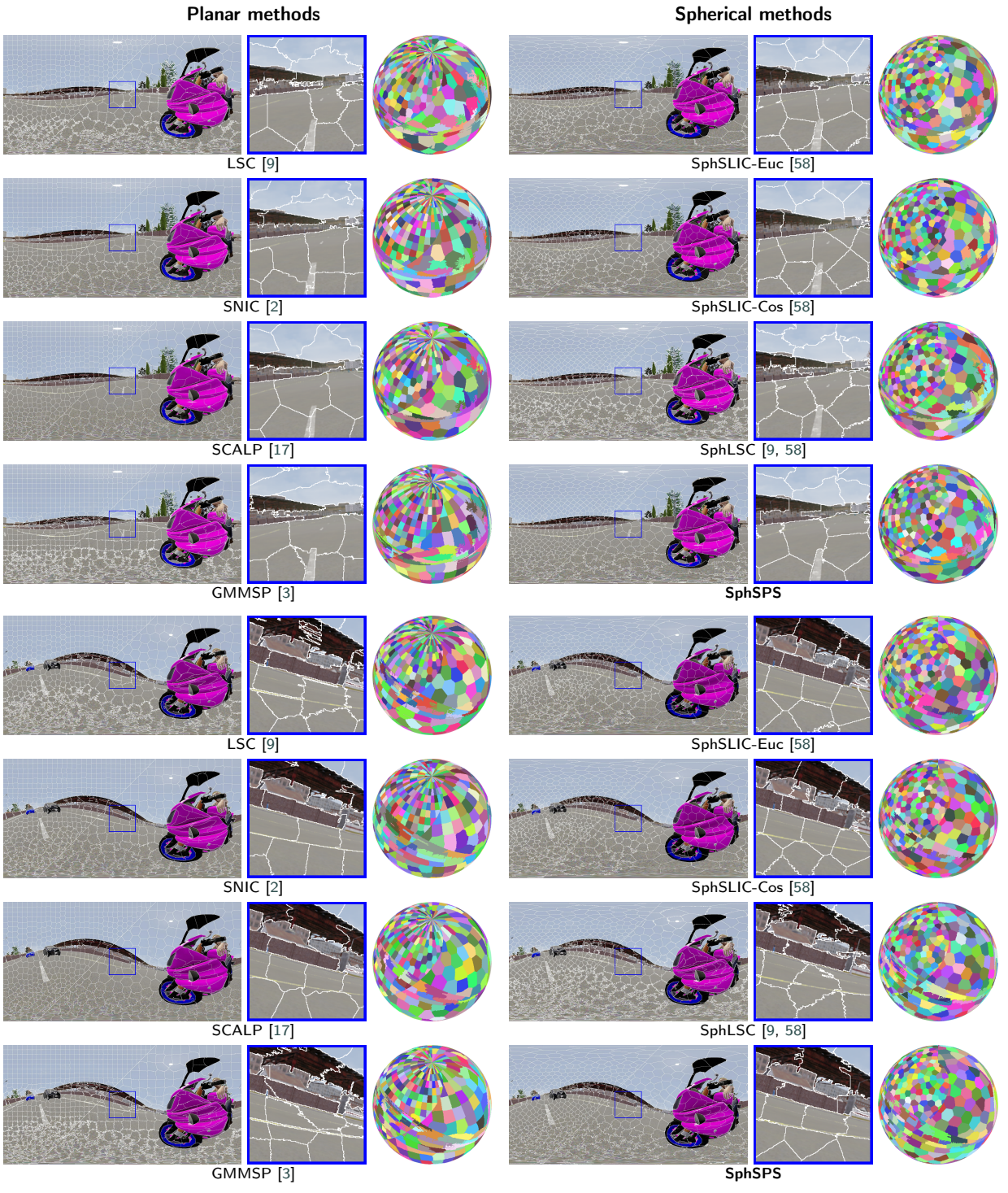


Figure 19: Visual comparison between SphSPS and the best planar and spherical (underlined) state-of-the-art methods on Omniscape images, for two superpixel numbers $K = 1200$ (top-left) and $K = 400$ (bottom right). SphSPS produces regular spherical superpixels with smooth boundaries that adhere well to the image contours

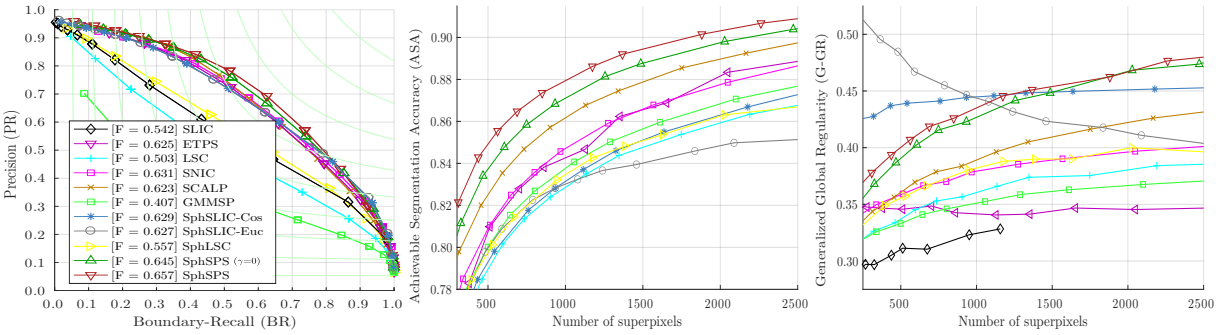


Figure 20: Quantitative comparison on the noisy PSD images [45] (addition of a white Gaussian noise of variance 20), on PR/BR, ASA and G-GR metrics of the proposed SphSPS method to the state-of-the-art ones

Table 1
Comparison to the state-of-the-art methods on initial|noisy PSD images [45], on the metrics presented in Section 4.1.2. The ASA (21) and G-GR (13) results are given for $K = 1500$ superpixels, and CD results for BR = 0.8 (19). Blue (bold) and red (underlined) respectively indicate best and second results. Some results for the SLIC method on noisy images are estimated by a linear fitting due to an insufficient number of generated superpixels

Method	F (20) ↑	CD/BR (19) ↓	ASA (21) ↑	G-GR (13) ↑
SLIC [1]	0.684|0.542	0.123|0.156	0.883|0.789	0.438|0.340
ETPS [56]	0.695|0.625	0.118|0.194	0.881|0.866	0.426|0.344
LSC [9]	0.668|0.503	0.130|0.313	0.890|0.848	0.407|0.374
SNIC [2]	0.672|0.631	0.125|0.137	0.884|0.866	0.405|0.389
SCALP [17]	0.688|0.623	0.114|**0.130**	0.887|0.879	0.451|0.409
GMMSP [3]	0.681|0.407	0.129|0.332	0.893|0.856	0.375|0.361
SphSLIC-Euc [58]	0.656|0.629	0.130|0.142	0.884|0.840	0.495|0.423
SphSLIC-Cos [58]	0.656|0.627	0.129|0.141	0.884|0.851	0.501|0.449
SphLSC [9, 58]	0.683|0.557	0.103|0.223	0.895|0.851	0.469|0.390
SphSPS (γ=0)	0.706|0.645	0.112|0.145	0.895|0.888	**0.519**|0.449
SphSPS	**0.710**|**0.657**	**0.111**|0.141	**0.899**|**0.894**	0.518|**0.454**

on the shortest path, SphSPS obtains significantly higher accuracy in only 2.30s per image, that is faster than existing spherical approaches [58].

Hence, the proposed optimizations in Section 2.3.3, enable to consider the large number of pixels contained into the shortest path, while only increasing the processing time by less than a factor 3. Finally, with basic multi-threading implementation, we easily reduce the processing time of SphSPS to 0.7s per image.

Further comparison of processing times would necessitate to consider the differences of implementation and optimization to reflect the computational potential of each method [41]. Nevertheless, since our method uses the same K-means clustering framework as SLIC [1], it is very likely to be able to perform in real-time, as other works have already proposed such GPU implementations [31, 10, 4].

5. Conclusion

In this work, we proposed a new paradigm to compute accurate superpixels directly in the 3D acquisition space considering the distorsions induced by the spherical projection to the planar image space. To this end, we generalized the shortest path approach to extract features between a pixel and a superpixel barycenter. To consider the color and contour feature information on such path enables to increase the segmentation performance and the regularity of the generated regions. We especially showed that respecting the acquisition space geometry enables to more accurately capture the image objects.

To evaluate the performance of the proposed SphSPS method, we first addressed the limitation of the existing spherical regularity measure by introducing a generalized regularity metric measuring the spatial convexity and consistency in the 3D spherical space. We extensively compared SphSPS to the state-of-the-art methods to demonstrate its superior performance on both natural and synthetic 360^o images, robustness to noise and low computational time.

Since joint border accuracy with high regularity in the acquisition space are crucial for relevant display and processing of neighboring relationships in computer vision pre-processing, we truly believe that the proposed work can be of high interest to the community. Future works will focus on extending our method to other acquisition spaces and omnidirectional video acquisitions.

A. Comparison of proposed G-GR and standard COM metrics

In Figure 22, we report the evaluation of methods on our previous implementation of the G-GR metric [14], and on the standard compactness measure COM (13), adapted to the spherical space in [58]. As discussed in Section 4.3, the proposed version of the G-GR metric enables to clearly differentiate spherical from planar methods, *i.e.*, with a significant number of superpixels, all spherical methods obtain higher regularity measure. Our previous implementation [14] uses a downscaling instead of a concave hull to obtain the final 2D projection of spherical superpixel shapes. It appears to be more impacted by fuzzy borders, so that the SphLSC method that produces very irregular spherical superpixels (see Figures 17, 18 and 19), is measured less regular than the planar SCALP method.

We also observe that the COM metric even fails at differentiating planar and spherical methods. For instance, the SCALP method [17] gets higher results than the SphSLIC method [58]. It may come from the non robustness of this criteria, which is only seen as a circularity notion and does not take into account the consistency of shape within the decomposition, while the proposed G-GR metric (18), evaluates the regularity of shapes and their consistency according to the acquisition space.

Finally, to further demonstrate the relevance of G-GR over the COM metric, we compute the correlation between the measured regularity and the number of generated superpixels. As the number of superpixel increases, their shape is very likely to be more regular, since they less have to stretch to gather homogeneous pixels. Hence, the regularity

is supposed to increase with the superpixel scale, *i.e.*, the number of superpixels. For all the compared methods in Section 4, on the PSD dataset, we report an average correlation between the number of superpixels and the regularity of 0.3725 for COM and of 0.6518 for the proposed G-GR.

Acknowledgments

We would like to thank Ahmed Rida Sekkat, Yohan Dupuis, Pascal Vasseur and Paul Honeine for sharing some images and their annotations in the frame of the Omniscape projet [39].

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurélien Lucchi, Pascal Fua, and Sabine Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34:2274–2282, 2012.

[2] Radhakrishna Achanta and Sabine Süsstrunk. Superpixels and polygons using simple non-iterative clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4895–4904, 2017.

[3] Zhihua Ban, Jianguo Liu, and Li Cao. Superpixel segmentation using gaussian mixture model. *IEEE Transactions on Image Processing*, 27(8):4105–4117, 2018.

[4] Zhihua Ban, Jianguo Liu, and Jeremy Fouriaux. GLSC: LSC superpixels at over 130 FPS. *Journal of Real-Time Image Processing*, 14(3):605–616, 2018.

[5] Åke Björck. Solving linear least squares problems by gram-schmidt orthogonalization. *BIT Numerical Mathematics*, 7(1):1–21, 1967.

[6] Jack E Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems Journal*, 4(1):25–30, 1965.

[7] Pierre Buyssens, Matthieu Toutain, Abderrahim Elmoataz, and Olivier Lézoray. Eikonal-based vertices growing and iterative seeding for efficient graph-based segmentation. In *IEEE International Conference on Image Processing*, pages 4368–4372, 2014.

[8] Ricardo Cabral and Yasutaka Furukawa. Piecewise planar and compact floorplan reconstruction from images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 628–635, 2014.

[9] Jiansheng Chen, Zhengqin Li, and Bo Huang. Linear spectral clustering superpixel. *IEEE Transactions on Image Processing*, 26:3317–3330, 2017.

[10] Kang-Sun Choi and Ki-Won Oh. Subsampling-based acceleration of simple linear iterative clustering for superpixel segmentation. *Computer Vision and Image Understanding*, 146:1–8, 2016.

[11] Thiago LT da Silveira, Adriano Q de Oliveira, Marcelo Walter, and Cláudio R Jung. Fast and accurate superpixel algorithms for 360º images. *Signal Processing*, 189, 2021.

[12] Amal Farag, Le Lu, Holger R Roth, Jiamin Liu, Evrim Turkbey, and Ronald M Summers. A bottom-up approach for pancreas segmentation using cascaded superpixels and (deep) image patch labeling. *IEEE Transactions on Image Processing*, 26(1):386–399, 2016.

[13] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[14] Rémi Giraud, Rodrigo Borba Pinheiro, and Yannick Berthoumieu. Generalized shortest path-based superpixels for accurate segmentation of spherical images. In *International Conference on Pattern Recognition*, 2020.

[15] Rémi Giraud, Vinh-Thong Ta, Aurélie Bugeau, Pierrick Coupé, and Nicolas Papadakis. SuperPatchMatch: an algorithm for robust correspondences using superpixel patches. *IEEE Transactions on Image Processing*, 26(8):4068–4078, 2017.

[16] Rémi Giraud, Vinh-Thong Ta, and Nicolas Papadakis. Evaluation framework of superpixel methods with a global regularity measure. *Journal of Electronic Imaging*, 26(6), 2017.

[17] Rémi Giraud, Vinh-Thong Ta, and Nicolas Papadakis. Robust superpixels using color and contour features along linear path. *Computer Vision and Image Understanding*, 170:1–13, 2018.

[18] Stephen Gould, Jiecheng Zhao, Xuming He, and Yuhang Zhang. Superpixel graph label transfer with learned distance metric. In *European Conference on Computer Vision*, pages 632–647, 2014.

[19] Mikhael Gromov et al. Filling Riemannian manifolds. *Journal of Differential Geometry*, 18(1):1–147, 1983.

[20] Varun Gulshan, Carsten Rother, Antonio Criminisi, Andrew Blake, and Andrew Zisserman. Geodesic star convexity for interactive image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3129–3136, 2010.

[21] Varun Jampani, Deqing Sun, Ming-Yu Liu, Ming-Hsuan Yang, and Jan Kautz. Superpixel sampling networks. In *European Conference on Computer Vision*, 2018.

[22] Charles FF Karney. Algorithms for geodesics. *Journal of Geodesy*, 87(1):43–55, 2013.

[23] Se-Ho Lee, Won-Dong Jang, and Chang-Su Kim. Tracking-by-segmentation using superpixel-wise neural network. *IEEE Access*, 2018.

[24] Alex Levinshtein, Adrian Stere, Kiriakos N Kutulakos, David J Fleet, Sven J Dickinson, and Kaleem Siddiqi. Turbopixels: fast superpixels using geometric flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2290–2297, 2009.

[25] Jiaying Liu, Wenhan Yang, Xiaoyan Sun, and Wenjun Zeng. Photo stylistic brush: robust style transfer via superpixel-based bipartite graph. *IEEE Trans. on Multimedia*, 20(7):1724–1737, 2017.

[26] Ming-Yu Liu, Oncel Tuzel, Srikumar Ramalingam, and Rama Chellappa. Entropy rate superpixel segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2097–2104, 2011.

[27] Vaïa Machairas, Matthieu Faessel, David Cárdenas-Peña, Théodore Chabardes, Thomas Walter, and Etienne Decencière. Waterpixels. *IEEE Transactions on Image Processing*, 24(11):3707–3716, 2015.

[28] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3061–3070, 2015.

[29] Kensuke Nakamura and Byung-Woo Hong. Hierarchical image segmentation via recursive superpixel with adaptive regularity. *Journal of Electronic Imaging*, 26(6), 2017.

[30] Peer Neubert and Peter Protzel. Superpixel benchmark and comparison. In *Forum Bildverarbeitung*, pages 1–12, 2012.

[31] Peer Neubert and Peter Protzel. Compact watershed and preemptive SLIC: On improving trade-offs of superpixel segmentation algorithms. In *International Conference on Pattern Recognition*, pages 996–1001, 2014.

[32] Shaul Oron, Aharon Bar-Hillel, Dan Levi, and Shai Avidan. Locally orderless tracking. *International Journal of Computer Vision*, 111(2):213–228, 2015.

[33] Robert Osserman et al. The isoperimetric inequality. *Bulletin of the American Mathematical Society*, 84(6):1182–1238, 1978.

[34] Tanu Priya, Saurabh Prasad, and Hao Wu. Superpixels for spatially reinforced bayesian classification of hyperspectral images. *IEEE Geoscience and Remote Sensing Letters*, 12(5):1071–1075, 2015.

[35] Matthias Reso, Jörn Jachalsky, Bodo Rosenhahn, and Jörn Ostermann. Temporally consistent superpixels. In *IEEE International Conference on Computer Vision*, pages 385–392, 2013.

[36] Antonio Rubio, LongLong Yu, Edgar Simo-Serra, and Francesc Moreno-Noguer. BASS: Boundary-aware superpixel segmentation. In *International Conference on Pattern Recognition*, 2016.

[37] Ken Sakurada and Takayuki Okatani. Change detection from a street image pair using CNN features and superpixel segmentation. In *British Machine Vision Conference*, pages 61–1, 2015.

[38] Alexander Schick, Mika Fischer, and Rainer Stiefelhagen. Measuring and evaluating the compactness of superpixels. In *International Conference on Pattern Recognition*, pages 930–934, 2012.

[39] Ahmed Rida Sekkat, Yohan Dupuis, Pascal Vasseur, and Paul Honeine. The OmniScape Dataset. In *International Conference on Robotics and Automation (ICRA)*, 2020.

[40] Ken Shoemake. Animating rotation with quaternion curves. In *International Conf. on Computer Graphics and Interactive Techniques*, pages 245–254, 1985.

[41] David Stutz, Alexander Hermans, and Bastian Leibe. Superpixels: An evaluation of the state-of-the-art. *Computer Vision and Image Understanding*, 166:1–27, 2016.

[42] Richard Swinbank and R James Purser. Fibonacci grids: A novel approach to global modelling. *Quarterly Journal of the Royal Meteorological Society: A journal of the atmospheric sciences, applied meteorology and physical oceanography*, 132(619):1769–1793, 2006.

[43] Joseph Tighe and Svetlana Lazebnik. SuperParsing: scalable nonparametric image parsing with superpixels. In *European Conference on Computer Vision*, pages 352–365, 2010.

[44] Roy Uziel, Meitar Ronen, and Oren Freifeld. Bayesian adaptive superpixel segmentation. In *IEEE International Conference on Computer Vision*, 2019.

[45] Liang Wan, Xiaorui Xu, Qiang Zhao, and Wei Feng. Spherical superpixels: benchmark and evaluation. In *Asian Conference on Computer Vision*, pages 703–717, 2018.

[46] Peng Wang, Gang Zeng, Rui Gan, Jingdong Wang, and Hongbin Zha. Structure-sensitive superpixels via geodesic distance. *International Journal of Computer Vision (IJCV)*, 103(1):1–21, 2013.

[47] Hongzhi Wang and Paul A Yushkevich. Multi-atlas segmentation without registration: a supervoxel-based approach. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 535–542, 2013.

[48] Xianshun Wang, Dongchen Zhu, Yanqing Liu, Xiaoqing Ye, Jiamao Li, and Xiaolin Zhang. Semflow: Semantic-driven interpolation for large displacement optical flow. *IEEE Access*, 7:51589–51597, 2019.

[49] Tien-Tsin Wong, Wai-Shing Luk, and Pheng-Ann Heng. Sampling with Hammersley and Halton points. *Journal of Graphics Tools*, 2(2):9–24, 1997.

[50] Jianxiong Xiao, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Recognizing scene viewpoint using panoramic place representation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2695–2702, 2012.

[51] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *IEEE International Conference on Computer Vision*, pages 1395–1403, 2015.

[52] Fengting Yang, Qian Sun, Hailin Jin, and Zihan Zhou. Superpixel segmentation with fully convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13964–13973, 2020.

[53] Hao Yang and Hui Zhang. Efficient 3D room shape recovery from a single panorama. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5422–5430, 2016.

[54] Kailun Yang, Xinxin Hu, Yicheng Fang, Kaiwei Wang, and Rainer Stiefelhagen. Omnisupervised omnidirectional semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[55] Kailun Yang, Xinxin Hu, and Rainer Stiefelhagen. Is context-aware CNN ready for the surroundings? panoramic semantic segmentation in the wild. *IEEE Transactions on Image Processing*, 2021.

[56] Jian Yao, Marko Boben, Sanja Fidler, and Raquel Urtasun. Real-time coarse-to-fine topologically preserving segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2947–2955, 2015.

[57] Yongxia Zhang, Xuemei Li, Xifeng Gao, and Caiming Zhang. A simple algorithm of superpixel segmentation with boundary constraint. *IEEE Transactions on Circuits and Systems for Video Technology*, (99), 2016.

[58] Q. Zhao, F. Dai, Y. Ma, L. Wan, J. Zhang, and Y. Zhang. Spherical superpixel segmentation. *IEEE Trans. on Multimedia*, 20(6):1406–1417, 2018.

[59] Denis Zorin and Alan H Barr. Correction of geometric perceptual distortions in pictures. In *International Conf. on Computer Graphics and Interactive Techniques*, pages 257–264, 1995.

Rémi Giraud received the M.Sc. in telecommunications at ENSEIRB-MATMECA, an engineer school member of Bordeaux Polytechnic Institute, France, in 2014. Since 2018, he is an associate professor at ENSEIRB-MATMECA in the Electronic department. He is also a researcher at the IMS laboratory in the Signal and Image group.

Rodrigo Pinheiro graduated in Electrical Engineering in University of Brasília, in 2021. He received the M.Sc. in Electronical Engineering at ENSEIRB-MATMECA, and the M.Sc. in Image and Signal Processing at University of Bordeaux, in 2020. Since 2021 he is a PhD Student at InterDigital, Inc. and at L2S, CentraleSupélec, France.

Yannick Berthoumieu received his Ph.D. degree in 1996 from the University of Bordeaux (UB). Since 2007 he is a full Professor of the Bordeaux Polytechnic Institute. His research activities cover areas such as statistical image and video processing in machine learning, for classification, segmentation, super-resolution and inversion algorithms.

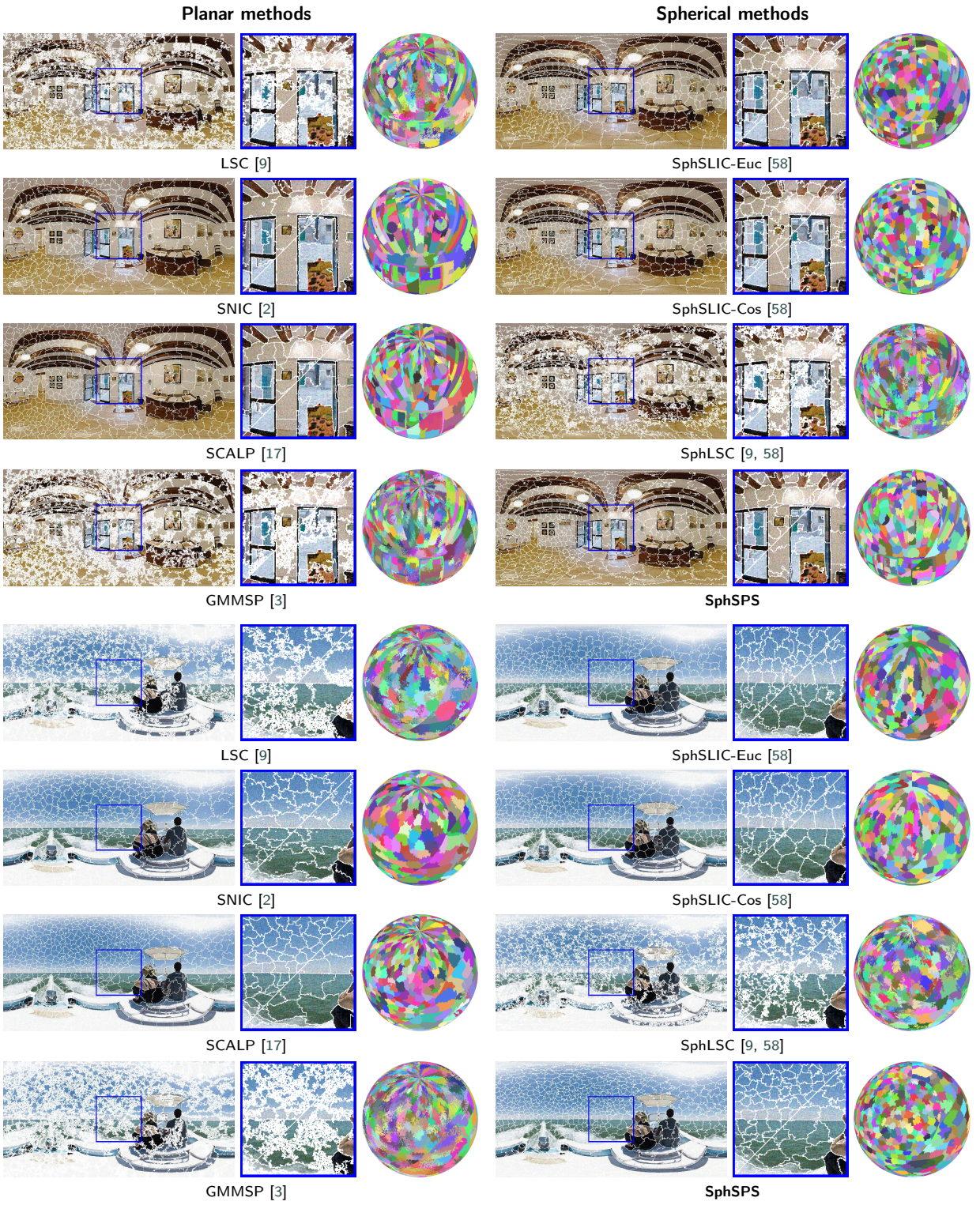


Figure 21: Visual comparison between SphSPS and the best planar and spherical (underlined) state-of-the-art methods on noisy PSD images, for two superpixel numbers $K = 1200$ (top-left) and $K = 400$ (bottom right). The compared methods may generate very inaccurate superpixels with noisy borders, while SphSPS remains robust to noise and produces regular superpixels with smooth boundaries that adhere well to the image contours

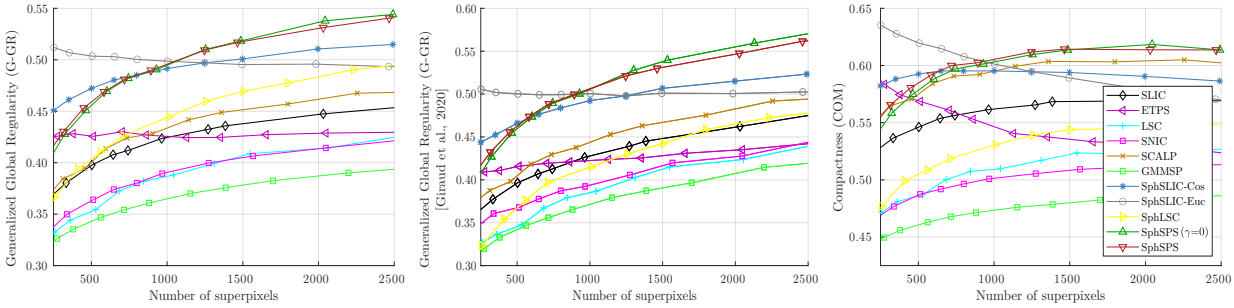


Figure 22: Comparison between proposed G-GR (left), our previous implementation of the G-GR metric [14] (center) and the spherical compactness COM [58] metric (right) on the PSD dataset. The proposed G-GR is able to differentiate spherical from planar methods while being more correlated to the number of generated superpixels