Gap-Sensitive Segmentation and Restoration of Digital Images

A. Sobiecki [1] and A. Jalba [2] and D. Boda [3] and A. Diaconeasa [3] and A. Telea [1,3]

[1] Johann Bernoulli Institute, University of Groningen, the Netherlands
[2] Department of Mathematics and Computer Science, University of Eindhoven, the Netherlands
[3] University of Medicine and Pharmacy "Carol Davila", Bucharest, Romania

Abstract
Many methods exist for removing defects such as gaps, cracks, and disconnections from digital shapes. However, most such methods have several limitations, such as removing both erroneous and important shape details, or requiring non-trivial effort from the end user in the form of manual delineation or parameter setting. In this paper, we propose a technique for removing defects such as internal gaps and cracks from 2D and 3D digital shapes. For this, we first classify gaps as boundary detail (to be preserved) and interior errors (to be removed), based on a heuristic that uses the gap position with respect the medial axis of the simplified shape. Next, we remove error gaps using an efficient distance-based filling. We illustrate our method on robust segmentation and hair removal tasks for skin imaging, and compare our results with a number of relevant techniques in this area.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

Reconstruction of shapes missing internal information serves a wide range of applications, such as repairing scans of deteriorated images by closing holes, improving shape recognition and shape matching, and connecting shapes that are broken into pieces [BBC*01, JT03, CPT04, CDD*04, Lie03, VCBS03]. Digital shapes missing internal information can often be filled with morphological operators as well as automatically or manually with inpainting techniques. However, morphological operations cannot discriminate between locally identical, but globally different, details, such as gaps close or on the shape boundary (which should not be filled, if we want to preserve boundary detail), and gaps deeper in the shape (which may need to be filled). Separately, inpainting requires extra work to select the areas to be inpainted, which requires manual effort or more involved, and thus more sensitive, image-analysis algorithms [CPT04, Tel04, LNG*97].

We propose a technique to detect and reconstruct 2D and 3D digital shape that lack some internal information, which we generically call 'gaps', while guaranteeing that detail information present on the apparent shape boundary is kept. For this, we first classify gaps into *detail* (that should be kept) and *errors* (that should be filled) using a global approach, based on the gap position with respect to the skeleton of a blurred version of the shape. Next, we fill error gaps using the medial axis transform associated to this skeleton. The method generically works for 2D and 3D digital shapes, using respectively 2D and 3D surface-skeletons, and is fast, simple to implement, and easy to use. We present applications for robust detail-preserving image segmentation and hair removal for dermatological images, and compare our method with several segmentation and restoration methods in the same field.

This paper is organized as follows. In Sec. 2, we review the related work. Section 3 presents our proposal. Section 4 presents 2D and 3D shape restoration examples. Section 5 presents an application of our method to the field of dermato-imaging. Section 6 discusses our method. Section 7 concludes the paper.

2. Related work

Many algorithms to segment and reconstruct digital shapes have been proposed. While an exhaustive review of the huge body of work on digital shape restoration is beyond our scope, we review three well-known approaches on segmentation and restoration of digital shapes which relate to our goals.

Filters: Filtering techniques like the median, mean, Laplacian [GL12], and morphological operators like erode, dilate, open, and close [HR98] can restore digital shapes by eliminating small-scale gaps, and are fast and simple to implement. However, most such filters work *locally*, so they cannot discriminate between gaps deep inside the shape (which we may want to eliminate) and gaps close or on the apparent shape boundary (which we want to keep, as they are part of the border structure).

Image segmentation: A key part of medical imaging is the segmentation of shapes from grayscale or color images. For example, in dermatology, one wants to segment tumors from surrounding healthy skin. Preserving all details on the segmentation border, and in the same time removing small-scale gaps and cracks inside the tumor, is essential for further automated analyses of the segmented image [DHR01, FRK85, PHJ10]. Several such segmentation methods exist [CM02, PHJ10, vdZMT13, FSL04]. However, as we shall see later in Sec. 5, none of these methods

can fully comply with the above two requirements.

Inpainting: Digital inpainting helps in restoring damaged parts of an image, such as reconstructing scans of deteriorated images by removing scratches or stains, or creating artistic effects [BBC*01, CPT04, Tel04]. Inpainting works well for reconstructing shapes that miss inside information, giving better results than the filter techniques mentioned above, but requires prior segmentation of the damaged region that in turn requires manual effort or more complex algorithms. For example, the DullRazor technique uses inpainting to digitally remove hairs from dermatological skin images, to make these images suitable for automatic analysis for diagnosis [LNG*97]. Although this technique works automatically, it requires a quite complex algorithm to robustly detect the hairs to be inpainted. Improvements of this technique are presented in [KS11], in terms of segmenting hairs of different colors, [XQJM09], for hair detection using morphological operations, and [ACG11], for comparing different inpainting schemes applied to the segmented hairs.

Salience skeletons: Skeletons, or medial axes, are descriptors that encode both the geometry and topology of 2D and 3D shapes [FSL04, TvW02]. Together with their distance field to the shape boundary, they yield the so-called medial axis transform (MAT), which can be used to reconstruct shapes [SP09]. Simplifying the skeleton prior to reconstruction by using various importance metrics, such as the salience metric [TK01, Tel12], allows a multiscale reconstruction of shapes which removes small-scale (noise) details but keeps important details. However, eliminating gaps using such methods is difficult, as these cause complex topological changes in the skeleton.

3. Our Proposal

Summarizing, our major goals are to (a) eliminate thin and long gaps that (nearly) disconnect a shape into several parts. We call these *error* gaps. In the same time, we want to (b) keep all *details*, including concave indentations or gaps, present on the shape's apparent boundary.

For this, we propose a three-step process (see Fig. 1 top). Given a shape $\Omega \subset \mathbb{R}^{n=\{2,3\}}$ with boundary $\partial\Omega$, stored as a binary image (black=foreground, white=background), we first close *all* gaps of Ω, using morphological operations (Sec. 3.1). Next, we use the resulting image Ω_{oc} to classify gaps into errors and details, using a topological analysis of Ω_{oc} (Sec. 3.2). Finally, we use related topological mechanisms to fill gaps identified as errors in the previous step (Sec. 3.3). These three steps are detailed next.

3.1. Gap Closing

To close all gaps present in our input image Ω, we use classical morphological operations. In detail, given a so-called structuring element H, we consider the dilation of Ω by H, *i.e.*, the union of copies of H_x, the element H centered at all pixels $x \in \Omega$, *i.e.*

$$\Omega \oplus H = \bigcup_{x \in \Omega} H_p. \qquad (1)$$

Similarly, we define the erosion of Ω by H, which keeps only pixels $x \in \Omega$ where H_x fits inside Ω, *i.e.*

$$\Omega \ominus H = \{x \in \Omega | H_x \subseteq \Omega\}. \qquad (2)$$

Next, we define the opening of Ω as erosion followed by dilation, *i.e.*

$$\Omega \circ H = (\Omega \ominus H) \oplus H, \qquad (3)$$

and, analogously, the closing of Ω as dilation followed by erosion, *i.e.*

$$\Omega \bullet H = (\Omega \oplus H) \ominus H. \qquad (4)$$

If we use a disk structuring element H of radius ρ, the result of applying opening and closing, denoted $\Omega_{oc} = (\Omega \circ H) \bullet H$, will close *all* holes in Ω whose thickness is smaller than ρ. Additionally, we denote the result of applying closing and opening, by $\Omega_{co} = (\Omega \bullet H) \circ H$. Both Ω_{oc} and Ω_{co} will be used next for our error-hole removal, see Secs. 3.2 and 3.3.

3.2. Gap Classification

We now use the image Ω_{oc} to classify holes into errors and detail. For this, we first compute the skeleton $S(\Omega_{oc})$. For this, we define the distance transform $DT_{\partial\Omega} : \Omega \to \mathbb{R}^n_+$ of any shape Ω as

$$DT_{\partial\Omega}(x \in \Omega) = \min_{y \in \partial\Omega} \|x - y\|. \qquad (5)$$

The skeleton $S(\Omega)$, or medial axis, of Ω is next defined as

$$S(\Omega) = \{x \in \Omega \,|\, \exists f_1, f_2 \in \partial\Omega, f_1 \neq f_2,$$
$$\|x - f_1\| = \|x - f_2\| = DT_{\partial\Omega}(x)\}. \qquad (6)$$

Figure 1 b shows the shape Ω_{oc} (in black) for our test image in Fig. 1 a, and the skeleton $S(\Omega_{oc})$ (in white) for the same shape.

We now compute the fragments F of the skeleton $S(\Omega_{oc})$ that fall outside our input shape Ω, *i.e.*

$$F = \{x \in S(\Omega_{oc}) | x \notin \Omega\}. \qquad (7)$$

We now observe that points in F are inside the error gaps, but outside the detail gaps, of Ω. Let us explain this. As noted in Sec. 3.1, Ω_{oc} closes both error and detail gaps of Ω, by construction. Additionally, Ω_{oc} has a boundary that is smoother than Ω (see Fig. 1 b). More precisely, all details on $\partial\Omega$ whose curvature is larger than $1/\rho$ are replaced by the close operation (Eqn. 4) by circle arcs in 2D (and respectively spherical segments in 3D) of radius ρ. We know that branches in $S(\Omega_{oc})$ correspond to curvature maxima on $\partial\Omega_{oc}$ [SP09]. Since $\partial\Omega_{oc}$ is smoother than Ω, it follows that branches of $S(\Omega_{oc})$, thus also points in F, will never be located inside *boundary* gaps, or details, of $\partial\Omega$, since (1) these correspond to curvature minima along $\partial\Omega$, and (2) Ω_{oc} has an absolute curvature smaller than $\partial\Omega$. On the other hand, since the branches of $S(\Omega_{oc})$ are centered in the middle of the salient features of Ω_{oc} (by the definition of the skeleton, Eqn. 6), they will also be centered in the middle of the corresponding salient features of Ω (compare Figs. 1 b and a). This is so because the open-close operation that constructs Ω_{oc} from Ω uses a *circular* disk H, so it does not modify the local shape symmetry. Overall, it follows that points in F will be located in gaps of Ω which protrude *deep* inside this shape.

3.3. Error Gap Restoration

To close the error gaps identified by the skeleton subset F, we next proceed as follows. For each point $p \in F$, we find its closest skeleton point being in the input shape Ω

$$C(p) = \operatorname*{argmin}_{q \in S(\Omega_{oc}) \cap \Omega} \|p - q\| \qquad (8)$$
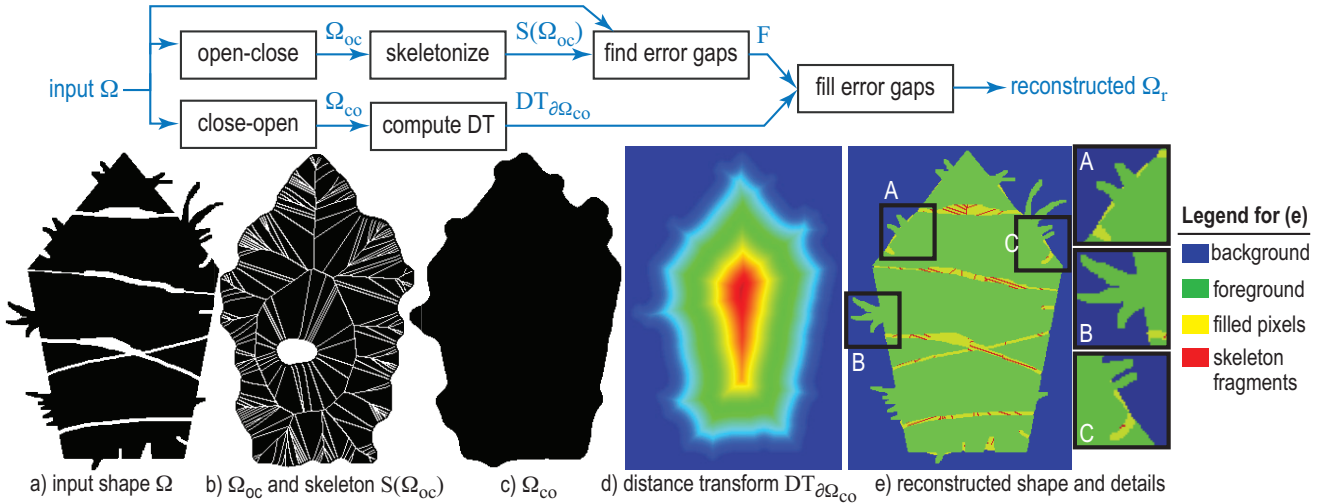
Figure 1: *Overview of proposed method. Top: algorithm pipeline. Bottom (a-e): Example on a test image. See Sec. 3.*

and then draw a foreground-disk with radius $DT_{\partial\Omega_{co}}(C(p))$ centered at p. This effectively fills the error gap containing p using the local shape thickness, which is equal to $DT_{\partial\Omega_{co}}(C(p))$. Let us explain this. First, we use here the distance transform of the shape Ω_{co} (see Fig. 1 d) obtained by the close-open operation, rather than the distance transform of Ω_{oc}, since the former first dilates, then erodes, the input shape. As such, Ω_{co} closes gaps better than Ω_{oc} (compare Fig. 1 c *vs* b). Thus, using $DT_{\partial\Omega_{co}}$ gives a better estimate of the apparent (filled) shape boundary within gaps than $DT_{\partial\Omega_{oc}}$. On the other hand, we use the skeleton of Ω_{oc} to detect error gaps, and initiate reconstruction from, rather than the skeleton of Ω_{co}, since Ω_{oc} does *not* close detail gaps (on the input boundary). If, in contrast, we used the skeleton of Ω_{co}, this skeleton would have branches that protrude outside Ω in boundary areas, and thus using F defined by Eqn. 7 would fill both error and detail gaps, which is undesired. Figure 2 details the above decision for a simple rectangle shape cut half-way by a vertical gap (Fig. 2 a). Images (b) and (c) show the results of the open-close and close-open operations, respectively. As visible, the close-open operation better fills the gap. Image (d) shows the reconstruction result if we used $DT_{\partial\Omega_{oc}}$. As visible, the gap is not well filled, since Ω_{oc} does not fill well the gap (image (b)). Image (e) shows our chosen reconstruction, where we use $DT_{\partial\Omega_{co}}$. The skeleton $S(\Omega_{oc})$, drawn in red, is of course identical. However, the disks drawn atop of the skeleton fragments F are now larger, since Ω_{co} is larger than Ω_{oc}, and thus, correspondingly, $DT_{\partial\Omega_{co}}(x) \geq DT_{\partial\Omega_{oc}}(x), \forall x \in F$.

Secondly, we note that $DT_{\partial\Omega_{co}}(p)$ is typically smaller than $DT_{\partial\Omega_{co}}(C(p))$, due to the effect of the close-open operation sequence. Hence, the gap filling done by this operation tends to 'shrink' the filled shape towards the middle of the gap. Hence, using $DT_{\partial\Omega_{co}}(C(p))$ instead of $DT_{\partial\Omega_{co}}(p)$, fills the gap by using a value which is much closer to the real shape thickness, and thus leads to a smoother reconstruction of the boundary of the filled shape across the error gap.

By the above procedure, error gaps which intersect the skeleton $S_{\Omega_{oc}}$ are thus eliminated. Figure 1 e shows the reconstructed shape $\Omega_r = \Omega \cup D$, where D is the set of pixels filled by the disk-drawing procedure outlined above. For clarity, we marked background pixels of Ω as blue, foreground Ω pixels as green, pixels in

D as yellow, and pixels in F as red. Intuitively, our reconstruction procedures implies that gaps which cut deep inside Ω, to be precise more than half of the local thickness, get filled. In particular, gaps which completely disconnect (cut) Ω, but which are removed in Ω_{oc} by the close operation, are guaranteed to be filled. In contrast, small superficial concavities or indentations of $\partial\Omega$ that do not intersect $S_{\Omega_{oc}}$, *i.e.* are less deep than half the local thickness of Ω, are never eliminated. This way, concave boundary details of Ω are kept (see insets in Fig. 1 e). Separately, note that the removal of *convex* details in Ω_{oc} (as compared to Ω, see Fig. 1 b *vs* a), due to the open operation (Eqn. 3), does not adversely affect our final result. Indeed, our gap filling only adds foreground pixels to Ω, but never eliminates pixels from it (see again insets in Fig. 1 e).

3.4. Implementation

For 2D skeleton extracting, we use the AFMM method [TvW02], which computes robust, centered, pixel-wide, and topologically correct skeletons for 2D shapes of up to 1024^2 pixels in subsecond time on a modern PC. For 3D surface skeletons, we use the IMA method [HR09], which shares the same desirable properties. Both the AFMM and IMA implicitly compute, besides skeletons, the exact Euclidean distance transform $DT_{\partial\Omega}$ of the input shape. This allows us to efficiently implement accurate dilation and erosion (Eqns. 1 and 2) by simply thresholding $DT_{\partial\Omega}$ with the desired radius of the disk structuring element H. Finally, we efficiently implement the disk-drawing filling in Sec. 3.3 by computing the distance transform DT_F of the set F and lower-thresholding it by the values $DT_{\partial\Omega_{co}}(C(p))$ for all points $p \in F$. Both AFMM and IMA methods are implemented in C++, and do not use parallelization. Overall, on a commodity 3.5 GHz PC, our entire pipeline takes subsecond time for 2D images up to 3000^2 pixels and a few seconds for 3D volumes up to 400^3 voxels.

4. Results

Figure 3 shows several 2D restoration examples for a set of synthetic shapes, on which gaps were created manually (a,d,j) or by luminance thresholding (g,m). As visible, our gap filling eliminates the complex internal gaps, but keeps the fine boundary details, including all boundary indentations. In contrast, if we were
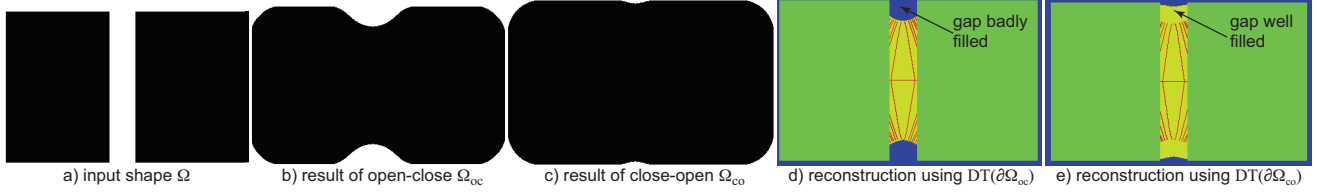
a) input shape Ω b) result of open-close Ω_{oc} c) result of close-open Ω_{co} d) reconstruction using DT($\partial\Omega_{oc}$) e) reconstruction using DT($\partial\Omega_{co}$)

Figure 2: *Effect of using distance transform of close-open shape Ω_{co} vs distance transform of open-close shape Ω_{oc}. See Sec. 3.3.*

to use a naive gap-filling by executing only an open-close operation sequence, the result would indeed fill most of the internal gaps, but also erase much of the (convex) boundary detail (images (b,e,h,k,n)). Images (m-r) show the effect of varying the structuring-element radius ρ (Sec. 3.1) for the input shape in image (m). Images (n) and (o) show, for illustration purposes, the open-closed shape Ω_{oc} and its skeleton $S(\Omega_{oc})$ respectively for the input image and the *rho* value for the result shown in image (q). As we increase ρ, larger internal gaps get progressively filled. However, fine-scale details on the apparent boundary of the input image stay preserved. As such, ρ can be effectively used to control the thickness of the internal shape gaps to be filled.

Figure 4 shows results for a set of binary shapes obtained from natural grayscale and color images via luminance thresholding. As expected, thresholding creates many disconnected components and/or holes and cracks within the perceived overall shapes. As for the synthetic images discussed earlier, open-close can fill most such gaps, but inherently destroys the boundary detail. In contrast, out method successfully removes gaps inside the apparent shape, but keeps most boundary detail.

Figure 5 shows a variation of our gap-filling technique. We start like in the previous cases, *i.e.*, we produce a binary segmentation (b) by luminance-thresholding of a grayscale CT brain image (a), taken from [Tel12]. The segmentation result shows significant noise and gaps that disconnect the apparent (black)foreground shape. Images (c-e) show the result of our gap-filling method. In contrast to the earlier examples (Figs. 3, 4), we use now the skeleton $S(\Omega_{co})$ instead of $S(\Omega_{oc})$. The effect is that more, and larger, gaps get filled, as we increase the structuring-element radius ρ. Additionally, instead of using the full skeleton $S(\Omega_{co})$, we now threshold it by eliminating branch end-fragments that correspond to fragments of the boundary $\partial\Omega_{co}$ shorter than τ pixels, using the skeleton-importance metric proposed in [TvW02], to which we refer for implementation details. This further smooths the boundary of the reconstructed shape (yellow pixels in Fig. 5). Overall, by increasing ρ and τ, we thus obtain a set of progressively simpler segmentations where larger holes are filled by smoother segments. However, as visible in images (c-e), small-scale convex boundary detail are still well preserved.

Finally, Fig. 6 shows our method applied to a 3D brain voxel dataset (a). Since we lacked an actual dataset with gaps, we generated these synthetically by performing several curved cuts (b). Image (c) shows the result of our method. As for the 2D examples, internal gaps are removed, while the brain surface detail is kept.

5. Applications for Skin Imaging

We next present several applications of our gap reconstruction technique in the field of skin imaging. Input images are dermatoscopic color images, of resolutions ranging from 500^2 to over
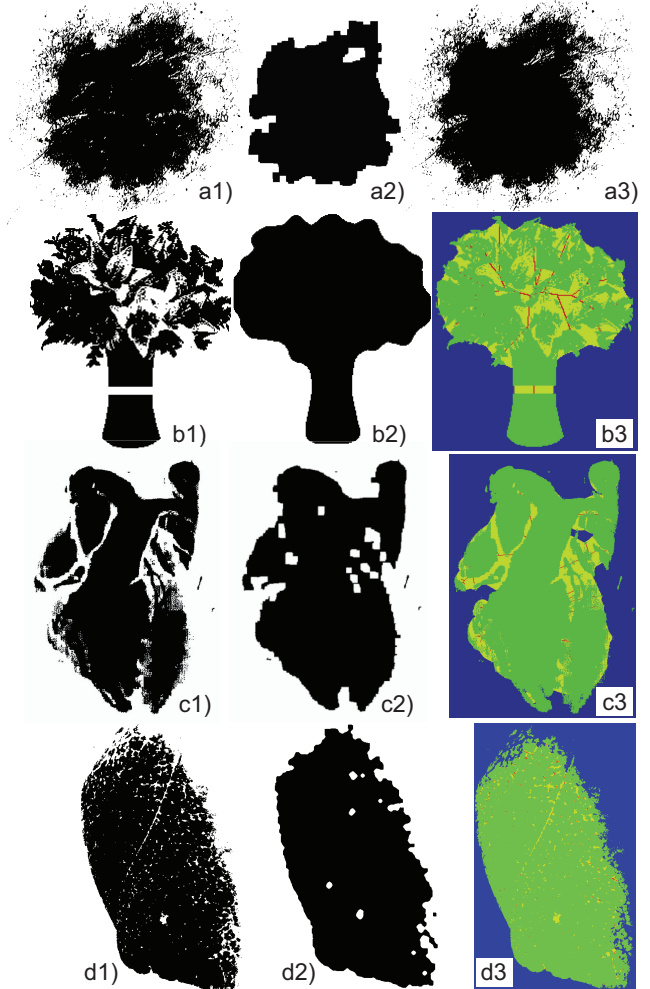

Figure 4: *Segmentation of natural 2D images: (1) Input shapes; (2) open-close operation; (3) proposed method. See Sec. 4.*

2500^2 pixels, showing skin tumors which can be either *naevi* (benign) or melanoma (malignant). Several techniques exist for the automatic pre-diagnosis of such tumors, based *e.g.* on the ABCD criteria [DHR01, FRK85, PHJ10]. However, to automatically evaluate such criteria, an accurate segmentation of the tumor from the surrounding skin is required. This is hard to do, as shown in Fig. 7, where we show the result of six known image segmentation methods on a typical dermatoscopic image(mean shift (MS) [CM02], gradient vector flow (GVF) [PHJ10], graph cuts (GC) [SM00], image foresting transform (IFT) [FSL04], level sets (LS) [LXGF10], and dense skeletons (DS) [vdZMT13]). Three types of problems occur. First, fuzzy tumor areas create strong ir-
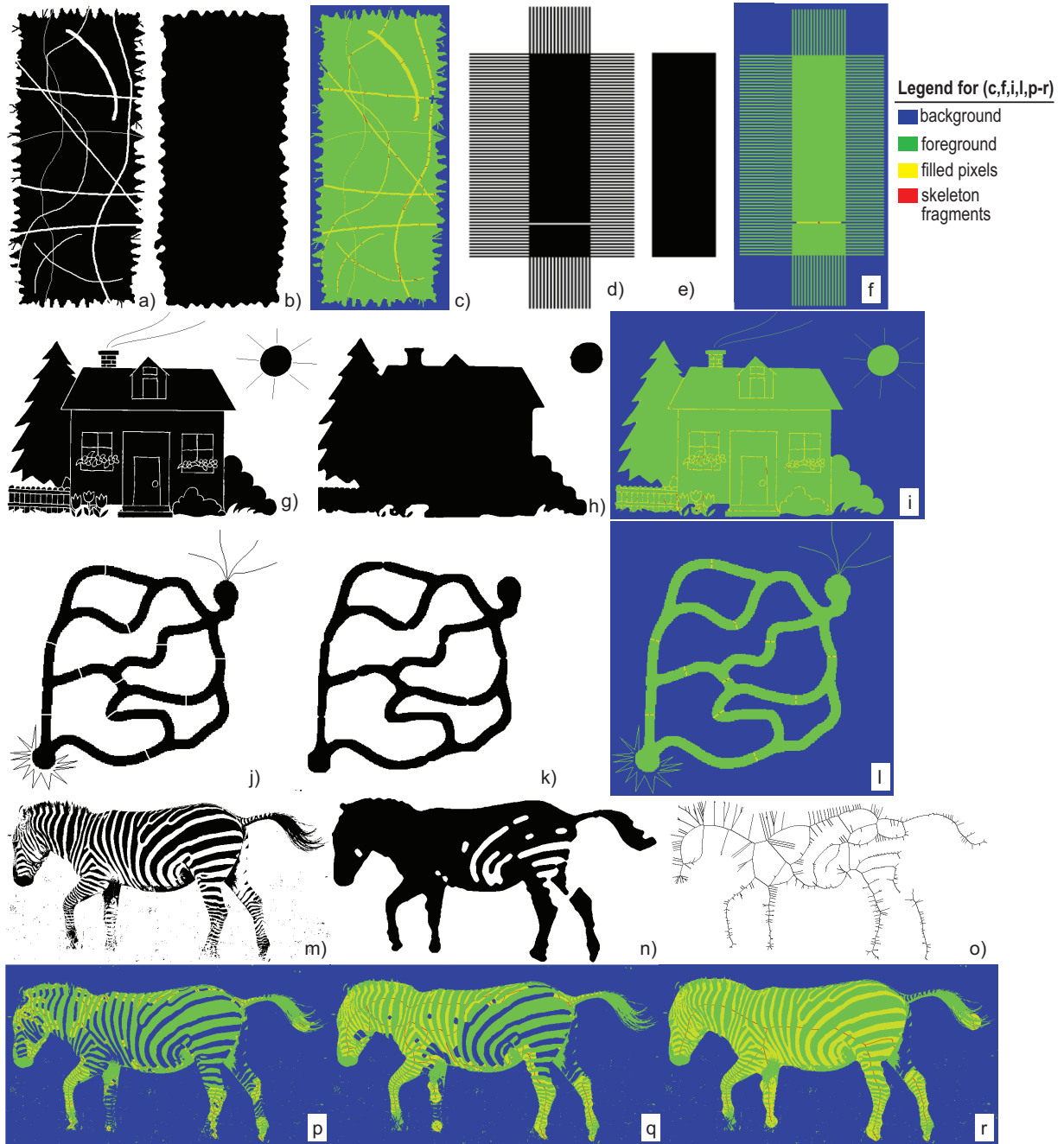
Figure 3: *Gap filling for a set of simple shapes. (a,d,g,j,m) Input shapes Ω. (b,e,h,k,n) Result of an open-close operation. (c,f,i,l,p-r) Gap-filling results, with blue=background pixels, green=foreground pixels, yellow=filled pixels, and red=skeleton pixels. See Sec. 4.*

regularities in the segmentation boundary (GC, MS). Methods with an in-built boundary smoothness remove such problems, but create too smooth boundaries missing image details (GVF). Both these issues create problems in evaluating the ABCD criteria [PHJ10]. Secondly, occluding hairs generate boundary artifacts (MS, LS). Finally, several methods are prone to oversegmentation (MS, GC, DS). All in all, this proves that segmentation of such images is a challenging task.

Figure 7 j shows the result of our method, applied to a luminance-based thresholding of the input skin image (Fig. 7 e). As visible, thresholding generates many holes, due to both inher-

ent color variation in the tumor, and to occluding hairs. As visible, our method generates smooth (but also detail-rich) boundaries, does not oversegment the image, and is not sensitive to occluding hairs. The gap-filling effectively removes the latter two issues, but does not remove the fine-scale detail present on the tumor boundary. Figure 7 b shows, for comparison, a manual segmentation performed by a dermatologist. While this segmentation is unavoidably not identical to ours, we notice that our result is, among the set of automatic techniques considered, the closest, both in shape and extent, to the manual segmentation.

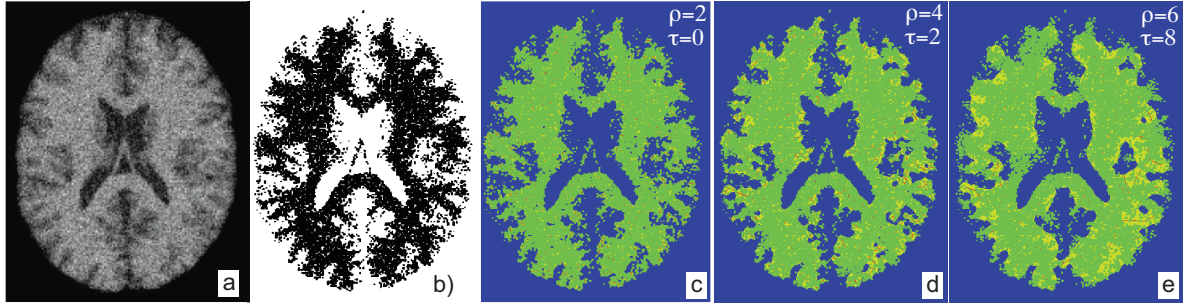Figure 8 shows the result of our method on five other skin-tumor

Figure 5: *Progressively smoother segmentations (c-e) of noisy brain image segmentation (b). See Sec. 4.*
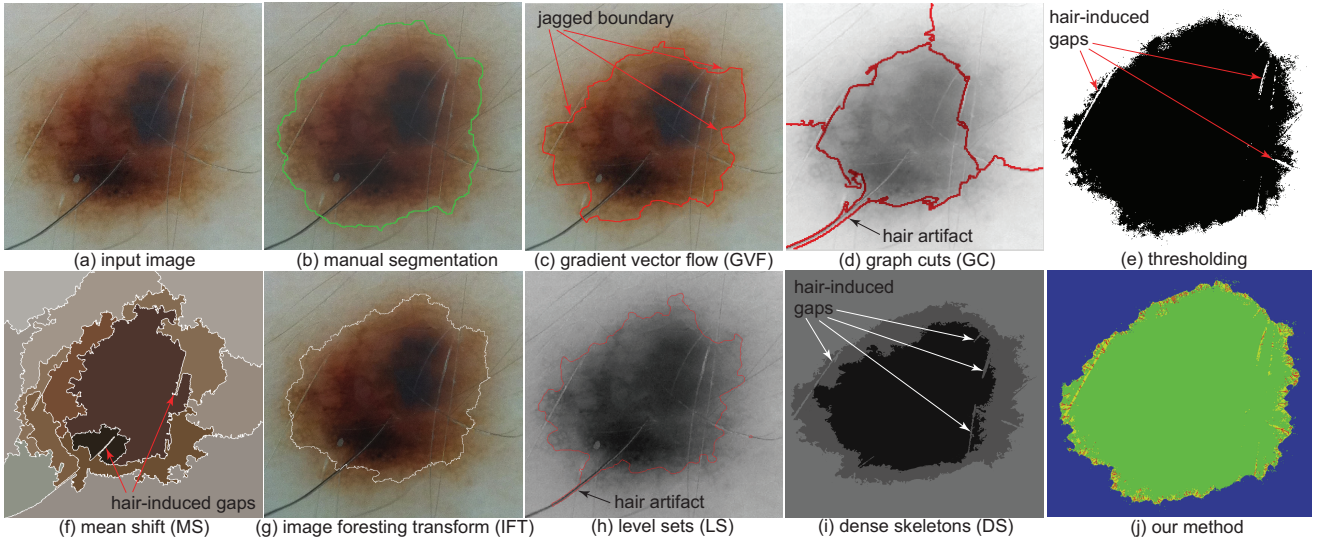

Figure 7: *Comparison of skin image segmentation – our method (j)* vs *seven other methods (c,d,e,f,g,h,i).*
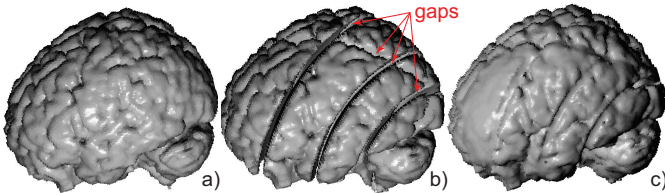

Figure 6: *Gap filling for 3D shapes. (a) Input shape. (b) Damaged shape showing several gaps. (c) Reconstruction by our method.*

images taken with different acquisition devices, of various resolutions, and showing widely different patterns that correspond to different types of skin diseases. As visible, our method improves the binary thresholding results by closing gaps inside the apparent tumor shape but keeping the tumor boundary details.

Finally, Figure 9 shows the use of our method for automatic hair removal in dermatoscopic images. The input image (a) show a very complex tumor shape, which is also covered by dense hair. Applying our technique on a luminance-thresholded image (b) yields the segmentation in (c). To remove hairs, we use $\Omega_r \setminus \Omega$ (see image (d)), *i.e.*, the difference between our result Ω_r (c) and the thresholded image Ω (b) as a mask for inpainting the input image using the method in [Tel04]. The result (f) shows that all internal hairs have been successfully removed while preserving the tumor

texture. Note that, for diagnostic image analysis, accurately segmenting the tumor *and* removing hairs inside the tumor only, is sufficient: Diagnostic analysis will next only run on the portion of the image inside the tumor, so all hairs (as well as healthy skin) outside the tumor are irrelevant. In contrast, the DullRazor method [LNG*97], with the software provided by the authors, one of the best-known hair-removal techniques in dermato-imaging, fails to remove most hairs (e), as it cannot robustly detect these, and is also considerably slower (16 seconds *vs* 0.6 seconds for our method on the platform mentioned in Sec. 3.2). Upon closer analysis, this is not surprising: DullRazor detects hairs using a contrast-based edge detector that works well for relatively separated constant-color hairs covering a low-contrast tumor of highly different luminance than the hairs. In our image, however, we have dense, variable-luminance, hairs that overlap a highly textured tumor, so this method fails.

For validation, we showed our skin-image segmentation and restoration results to a dermatology specialist, with over 6 working-experience years in dermato-oncology. We posed a set of questions pertaining qualitative aspects of our results, such as perceived correctness, relative quality with respect to other similar automatic methods, and relative quality with respect to manual segmentation. The test-set included over 30 images (not all present in this paper). The specialist responded very positively, pointing our that our segmentations are, in nearly all cases, superior in terms of boundary smoothness, detail preservation, and ease-of-use, to
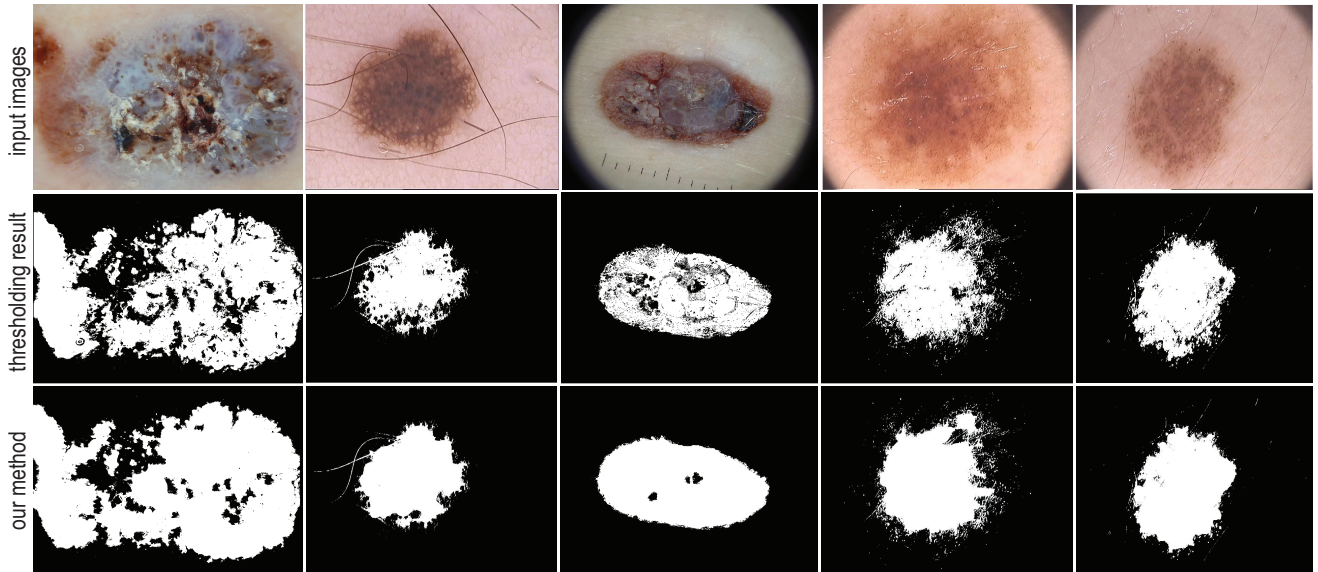
Figure 8: *Application to clean segmentation of five dermatoscopy skin images. See Sec. 5.*

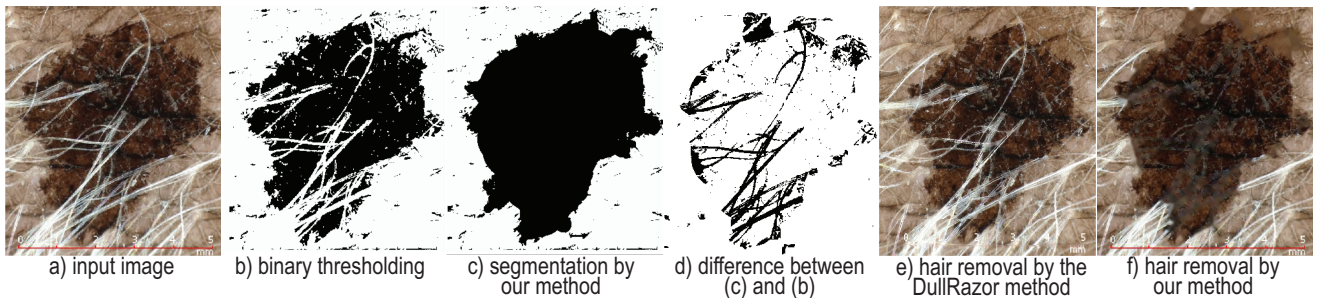| a) input image | b) binary thresholding | c) segmentation by our method | d) difference between (c) and (b) | e) hair removal by the DullRazor method | f) hair removal by our method |

Figure 9: *Automatic hair removal in complex skin tumor images. See Sec. 5.*

any known automatic method (though she indicated that manual segmentation can sometimes perform better in some fuzzy image areas). Additionally, our hair-removal method was found qualitatively better than manual alternatives and much better than DullRazor, for all complex images being tested, and equally good to these methods for the simple (low-hair-occlusion) test images. In particular, it was noted that our method has only two user-parameters (the luminance threshold level and gap-filling radius ρ), so it is much simpler to learn and use than other methods which expose more, and more complex, parameters.

6. Discussion

Below we discuss several aspects of our method.

Strengths: The main strength of our method is the ability to close gaps which appear *inside* the input binary shape, and in the same time keep both convex and concave detail present on the apparent boundary of the same shape. The method can handle well gaps of variable position, orientation, and thickness, as demonstrated by the examples shown in this paper. The single user-parameter to control is ρ, the maximal thickness of gaps to be closed, which has an intuitive meaning. Experiments done showed that our method can yield good-quality segmentations and restorations of dermatoscopic images, which are perceived to be better, and more

useful, by domain specialists.

Limitations: The key heuristic of our method is the classification of error gaps (to be filled) as being those which intersect the skeleton of a simplified (open-close) version Ω_{oc} of the input shape Ω. The main rationale behind this heuristic is that (a) open-close simplifies Ω by removing details but keeping its main structure, and its parameter ρ allows specifying the maximal thickness of gaps to be filled (*e.g.*, allows users to specify that large gaps are important details, so should not be filled); (b) hence, the skeleton of the simplified shape Ω_{oc} captures the main part-whole structure of the original Ω; (c) gaps in the original Ω that cut this skeleton affect thus more critically the 'structure' of Ω, and thus should be removed, than gaps far from the skeleton, which can be safely regarded as details of Ω. Clearly, there can exist application contexts where step (c) of our heuristic would fail. In such cases, our method would fill less gaps than desired. However, in the over 120 examples tested so far, we have noticed that our heuristic works as expected, *i.e.* discriminates between relevant gaps (far from the shape skeleton, and thus should be kept) and error gaps (which locally cut the shape more than half, and thus should be removed) in the desired way. However, we fully agree that our heuristic needs more testing before being able to state its value in a strong sense.

Comparison: In our presented examples, we make a number of

simplifications. First, we only use basic luminance thresholding for creating the input binary images for our gap-filling and restoration process. Clearly, more advanced techniques can be used. However, we chose a simple technique precisely to be able to demonstrate the added-value of our method on poor-quality input images. Secondly, the comparison against the six segmentation methods in Sec. 5 is surely limited, as more such methods exist. However, as stated, it is noteworthy that our (simpler) method performs qualitatively better than this range of very different segmentation methods. Thirdly, our inpainting examples only use a simple technique [TvW02]. We do this to clearly separate the inpainting effects from the added-value of our method. End applications can immediately swap our choice for more complex, and qualitatively better, inpainting, *e.g.* [CPT04, BBC*01]. Finally, we note that the choice of the AFMM [TvW02] and IMA [HR09] skeletons is important. One can use other 2D or 3D skeletonization methods, provided that these can produce correct multiscale, pixel/voxel-thin, centered, and connected, skeletons from any complex, noisy, disconnected shape. Unless skeletons have all above properties, the error-gap detection (Sec. 3.2) and gap filling (Sec. 3.3) would not correctly work.

Acknowledgments

7. Conclusion

We have presented a method for reconstruction of binary 2D and 3D shapes that miss internal information in the form of holes, disconnections, and cracks. In contrast to local filtering methods, which can remove such artifacts, but also smooth our relevant details on the shape boundary, our method can successfully remove these artifacts but fully preserve the shape boundary. To achieve this, we propose a heuristic to classify gaps in terms of their position to the shape skeleton, and next remove deep gaps which intersect this skeleton. We efficiently implement our method by means of distance transform and skeletonization algorithms for both the 2D and 3D cases. Finally, we present a concrete application of our technique for robust image segmentation and hair removal in dermatological applications, and compare our results with a number of known segmentation and one restoration technique in this field.

Future work can target a number of directions. Technique-wise, our method could be extended to the area of hole and crack filling in 3D surface meshes [Lie03, VCBS03]. Application-wise, we can adapt our method for the reconstruction of 3D scalar and/or vector fields, such as CT and MRI scans, by removal and restoration of low-quality and low-certainty areas [DMGL02]

References

[ACG11] ABBAS Q., CELEBI M., GARCIA I.: Hair removal methods: A comparative study for dermoscopy images. *Biomedical Signal Processing and Control 6* (2011), 395–404. 2

[BBC*01] BALLESTER C., BERTALMIO M., CASELLES V., SAPIRO G., VERDERA J.: Filling-in by joint interpolation of vector fields and gray levels. *IEEE Trans. Imag. Proc. 10*, 12 (2001), 1200–1211. 1, 2, 8

[CDD*04] CLARENZ U., DIEWALD U., DZIUK G., RUMPF M., RUSU R.: A finite element method for surface restoration with smooth boundary conditions. *CAGD 21*, 5 (2004), 427 – 445. 1

[CM02] COMANICIU D., MEER P.: Mean shift: A robust approach toward feature space analysis. *IEEE TPAMI 24*, 5 (2002), 603–619. 1, 4

[CPT04] CRIMINISI A., PEREZ P., TOYAMA K.: Region filling and object removal by exemplar-based image inpainting. *IEEE Trans. Imag. Proc. 13*, 9 (2004), 1200–1212. 1, 2, 8

[DHR01] DEVITA V., HELLMAN S., ROSENBERG S.: *Cancer: Principles & Practice of Oncology*. Lippincott Williams & Wilkins, 2001. 1, 4

[DMGL02] DAVIS J., MARSCHNER S., GARR M., LEVOY M.: Filling holes in complex surfaces using volumetric diffusion. In *Proc. 3D Data Processing Visualization and Transmission* (2002), pp. 428–438. 8

[FRK85] FRIEDMAN R., RIGEL D., KOPF A.: Early detection of malignant melanoma: The role of physician examination and self-examination of the skin. *CA Cancer J Clin 35*, 3 (1985), 130–151. 1, 4

[FSL04] FALCÃO A. X., STOLFI J., LOTUFO R. A.: The image foresting transform: Theory, algorithms, and applications. *IEEE TPAMI 26*, 1 (2004), 19–29. 1, 2, 4

[GL12] GUNTURK B., LI X.: *Image Restoration: Fundamentals and Advances*. Digital Imaging and Computer Vision. CRC Press, 2012. 1

[HR98] HEIJMANS H., ROERDINK J.: *Mathematical Morphology and its Applications to Image and Signal Processing*, vol. 12 of *Computational Imaging and Vision*. Kluwer, 1998. 1

[HR09] HESSELINK W., ROERDINK J.: Euclidean skeletons of digital image and volume data in linear time by the integer medial axis transform. *IEEE TPAMI 12*, 30 (2009), 2204–2217. 3, 8

[JT03] JIA J., TANG C.: Image repairing: Robust image synthesis by adaptive *n*D tensor voting. In *Proc. IEEE CVPR* (2003), pp. 643–650. 1

[KS11] KIANI K., SHARAFAT A.: E-shaver: An improved dullrazor for digitally removing dark and light-colored hairs in dermoscopic images. *Computers in Biology and Medicine 41* (2011), 139–145. 2

[Lie03] LIEPA P.: Filling holes in meshes. In *Proc. SGP* (2003), Eurographics, pp. 200–205. 1, 8

[LNG*97] LEE T., NG V., GALLAGHER R., COLDMAN A., MCLEAN D.: DullRazor: A software approach to hair removal from images. *Comput. Biol. Med. 27*, 6 (1997), 533–543. 1, 2, 6

[LXGF10] LI C., XU C., GUI C., FOX M. D.: Distance regularized level set evolution and its application to image segmentation. *IEEE TPAMI 19*, 12 (2010), 3243–3254. 4

[PHJ10] PAROLIN A., HERZER E., JUNG C.: Semi-automated diagnosis of melanoma through the analysis of dermatological images. In *Proc. SIBGRAPI* (Los Alamitos, CA:, 2010), IEEE Press, pp. 71–78. 1, 4, 5

[SM00] SHI J., MALIK J.: Normalized cuts and image segmentation. *IEEE TPAMI 22*, 8 (2000), 888–905. 4

[SP09] SIDDIQI K., PIZER S.: *Medial Representations: Mathematics, Algorithms and Applications*. Springer, 2009. 2

[Tel04] TELEA A.: An image inpainting technique based on the fast marching method. *J. Graphics Tools 9*, 1 (2004), 22–34. 1, 2, 6

[Tel12] TELEA A.: Feature preserving smoothing of shapes using saliency skeletons. In *Visualization in Medicine and Life Sciences* (2012), pp. 153–170. 2, 4

[TK01] TEK H., KIMIA B.: Boundary smoothing via symmetry transforms. *J. Math. Imag. Vis. 14* (2001), 211–223. 2

[TvW02] TELEA A., VAN WIJK J. J.: An Augmented Fast Marching Method for Computing Skeletons and Centerlines. In *Proc. VisSym* (2002), Springer, pp. 251–258. 2, 3, 4, 8

[VCBS03] VERDERA J., CASELLES V., BERTALMIO M., SAPIRO G.: Inpainting surface holes. In *Proc. ICIP* (2003), pp. 342–332. 1, 8

[vdZMT13] VAN DER ZWAN M., MEIBURG Y., TELEA A.: A dense medial descriptor for image analysis. In *Proc. IEEE VISAPP* (2013), pp. 133–140. 1, 4

[XQJM09] XIE F., QIN S., JIANG Z., MENG R.: PDE-based unsupervised repair of hair-occluded information in dermoscopy images of melanoma. *Comp. Med. Imag. Graph. 33* (2009), 275–282. 2