

A Virtually Continuous Representation of the Deep Structure of Scale-Space

Luigi Rocca and Enrico Puppo

Department of Informatics, Bio-engineering, Robotics and System Engineering
University of Genova - Via Dodecaneso 35 - 16146
Genova - ITALY {rocca|puppo}@dibris.unige.it

Abstract. The deep structure of scale-space of a signal refers to tracking the zero-crossings of differential invariants across scales. In classical approaches, feature tracking is performed by neighbor search between consecutive levels of a discrete collection of scales. Such an approach is prone to noise and tracking errors and provides just a coarse representation of the deep structure. We propose a new approach that allows us to construct a virtually continuous scale-space for scalar functions, supporting reliable tracking and a fine representation of the deep structure of their critical points. Our approach is based on a piecewise-linear approximation of the scale-space, in both space and scale dimensions. We present results on terrain data and range images.

Keywords: scale-space, multi-scale analysis, topology of scalar fields

1 Introduction

Since seminal works of Witkin [12] and Koenderink [8], scale-space methods have been greatly studied in the literature on computer vision and image processing - see [10] Section 4.4 for an updated bibliography - and later on in scientific visualization [1, 2, 6, 7]. The *deep structure* of the scale-space [9] captures the evolution of differential properties across scales, by tracking zero-crossings of differential invariants - most typically, the critical points of a scalar field. The importance of critical points is measured by their life time in the scale-space and helps identifying relevant features that subsume the structure of the signal.

The standard approach to scale-space is discrete: a scale-space consists of a collection (f_0, \dots, f_k) of subsequently filtered versions of an input signal $f = f_0$. Feature points are identified in each level f_i and the deep structure is extracted by tracking each feature across pairwise consecutive signals f_i, f_{i+1} : given a feature point in f_{i+1} , a neighborhood of its location is explored in f_i , searching for a feature of the same type; feature points of f_i that do not match any feature point in f_{i+1} are considered to end their life at scale i ; etc. Such a procedure is prone to false and missed matchings, and the matching process is greatly influenced by the granularity of sampling through scales.

In [11], a method was introduced to track the critical points of a scalar field, which follows the gradient of the field to detect correspondences across levels of

39 the scale-space. Albeit more robust than standard neighbor search, this approach
 40 is still discrete, as the analysis is again performed at the level of granularity of
 41 sampling in the scale dimension, with similar drawbacks.

42 In this work, we propose a new approach that provides a virtually continuous
 43 model of the scale-space and of its deep structure. We adopt a piecewise-linear
 44 representation of the input signal f , by discretizing its domain with a simplicial
 45 mesh (a triangle mesh for a bi-variate function) having vertices at the sample
 46 points. The connectivity structure of samples provides a straightforward local
 47 criterion to detect critical points. We also adopt a piecewise-linear representa-
 48 tion of the diffusion flow that generates the scale-space. Given the collection of
 49 snapshots (f_0, \dots, f_k) , we assume the flow to be linear between each pair f_i ,
 50 f_{i+1} . This allows us to easily track critical points across scales, because changes
 51 in the deep structure can occur only at a finite set of events, which are easy to
 52 find. The granularity of such events is much finer than sampling of snapshots.
 53 Tracking of critical points is exact in the context of this piecewise-linear approx-
 54 imation, while density of sampling in the scale dimension is just relevant to the
 55 approximation of diffusion flow, which is a non-linear process.

56 The trajectory of each critical point in the deep structure is encoded as a
 57 chain of edges of a mesh discretizing the domain of f ; in this model, velocity
 58 of displacement of each critical point along its trajectory is piecewise constant
 59 and it is encoded by storing times of transitions through vertices of the mesh
 60 on a continuous scale; bifurcations, corresponding to deletion/creation of pairs
 61 of critical points, occur at midpoints of edges of the mesh where trajectories
 62 end/start. With this data structure at hand, it is straightforward to perform a
 63 number of queries about the scale-space and its deep structure, such as: slicing
 64 the scale-space at any arbitrary scale; evaluating and visualizing the trajectories
 65 of all critical points; selecting critical points according to their life span; etc.

66 **2 Scale-space**

67 For the sake of simplicity, in the following we will deal just with the linear scale-
 68 space of a bi-variate scalar function defined on a 2D rectangular domain and
 69 sampled on a regular grid (such as images and digital elevation models). Most
 70 concepts introduced in this article, however, can be generalized in a straightfor-
 71 ward way to any kind of scale-space, for functions defined over general manifold
 72 domains, in higher dimensions, and with irregular sampling. Generalizations will
 73 be briefly discussed in Section 4.

74 Let $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ be a bi-variate scalar function with bounded range. For
 75 convenience, we assume that f is a Morse function, i.e., all critical points of f
 76 are isolated. The linear scale-space $F_f(x, y, t)$ of f is defined as the solution of
 77 the *heat equation*

$$\frac{\partial}{\partial t} F_f = \alpha \Delta F_f,$$

78 with initial condition $F_f(x, y, 0) = f(x, y)$, where Δ denotes the Laplace operator
 79 (with respect to the space variables x, y) and α is a constant term tuning the

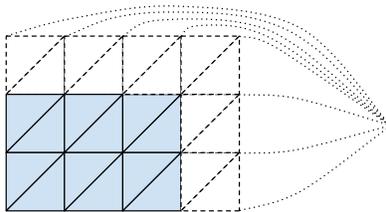


Fig. 1: Piecewise linear discretization of the domain (light blue triangles). Virtual points in the outer border (white triangles, dashed lines) correspond to a single point “at infinity” that completes the domain to a sphere-like topology.

80 speed of diffusion. So, scale-space is defined on a three dimensional domain, the
 81 first two dimensions referring to space. We will use interchangeably the words
 82 *scale* and *time* referring to the third dimension, as time comes from physical
 83 interpretation of the heat equation. Function F_f can be obtained either through
 84 a diffusion process starting at f , or, equivalently, by filtering f with Gaussian
 85 kernels of increasing variance, where variance is directly proportional to t .

86 Let p be a critical point of f , i.e., a maximum, or a minimum, or a saddle.
 87 Generally speaking, point p is displaced by the diffusion process, i.e.: it will
 88 possibly change its position for different values of t ; and eventually it may disap-
 89 pear by collapsing with another critical point p' . Collapses always involve critical
 90 points of different types, the only possibilities being either minimum-saddle, or
 91 maximum-saddle. In linear scale-space - as well as in most other existing scale-
 92 spaces - critical points may also appear in pairs at a time $t > 0$, with the same
 93 pairing rules. The tracking of a critical point p through its life in scale-space
 94 provides a *trajectory*, which is a continuous line in the 2D + time domain. The
 95 trajectory starts either at $t = 0$, or at time of birth of p ; and it either ends
 96 at time of collapse, or it extends to infinity. The *life span* of p is the interval
 97 spanned by the trajectory in time dimension, which provides a measure of how
 98 relevant p is across the different scales.

99 In a discrete representation, scale-space is sampled at a finite set of times
 100 ($t_0 = 0, t_1, \dots, t_k$) on a bounded domain D , which we will assume to be a
 101 rectangle. A straightforward discrete representation consists of a collection of
 102 snapshots (f_0, \dots, f_k), where each snapshot f_i is a discrete sampling of $F_f(\cdot, \cdot, t_i)$
 103 at the nodes of a regular $m \times n$ grid G_D over domain D , i.e., $f_i = \{f_i[r, c] \mid r =$
 104 $1, \dots, m, c = 1, \dots, n\}$. If p is a node of G_D at coordinates $[r, c]$, we will use the
 105 shorthand $f_i(p)$ to denote $f_i[r, c]$.

106 2.1 Piecewise-linear discretization of space

107 We adopt a piecewise-linear representation of D by subdividing it into triangles
 108 as follows: each sampling point of coordinates $[r, c]$ is connected to its neighbors
 109 to form triangles ($[r, c], [r, c + 1], [r + 1, c + 1]$) and ($[r, c], [r + 1, c + 1], [r + 1, c]$),
 110 as depicted in Figure 1. We extend the domain with a virtual outer border,

111 imposing the value of all samples of f_i in the outer border to be an arbitrarily
 112 large negative value (virtually, $-\infty$). In this way, all sampling points have exactly
 113 six neighbors; the extended domain can be seen as having the topology of a sphere
 114 (where all nodes in the outer border collapse to a single point); and the function
 115 has a global minimum in the outer border, at all scales.

116 A discrete function sampled at the nodes of G_D is thus extended to a
 117 piecewise-linear function interpolating the samples. With abuse of notation, in
 118 the following we will use the same symbols f_0, \dots, f_k to denote the piecewise-
 119 linear models of the sampled functions in the scale-space. Critical points of a
 120 function can occur only at sampling points:

- 121 – A minimum/maximum is a sample whose value is smaller/larger than the
 122 values of all its neighbors;
- 123 – A saddle is a sample such that, when its neighbors are traversed in cyclic ra-
 124 dial order, the number s of times that their values alternate between smaller
 125 and larger values with respect to the sample itself is larger than two. The
 126 index of the saddle is $s/2 - 1$: since each sampling point has exactly six
 127 neighbors, only 1- and 2-saddles may exist.

128 Note that in the smooth case, a Morse function admits just 1-saddles. Multiple
 129 saddles that arise in the piecewise-linear model correspond to different 1-saddles
 130 collapsing to the same point, and so they will be treated in our model. The
 131 Poincaré-Hopf index theorem guarantees that $n_M + n_m - n_s = 1$, where n_M
 132 is the number of maxima, n_m is the number of minima (excluding the virtual
 133 minimum in the outer border), and n_s is the number of saddles (where each
 134 2-saddle counts for two). Also note that a classical model based on either 4-
 135 or 8-connectedness of pixels would *not* provide a classification of critical points
 136 consistent with the theorem, thus hindering the correct pairwise simplification
 137 of critical points through scales (see Section 2.3).

138 **2.2 Piecewise-linear discretization of time**

139 Given the piecewise-linear snapshots f_0, \dots, f_k as defined above, let $p = [r, c]$
 140 be a node of G_D . We define a piecewise-linear approximation of $F_f(r, c, \cdot)$ inter-
 141 polating all samples $f_0(p), \dots, f_k(p)$. In this way, we may slice the scale-space
 142 in the continuum, at any time $t \in [t_0, t_k]$, obtaining a new snapshot f_t that is
 143 again a piecewise-linear function defined on the same triangular tiling of D . Let
 144 i be such that $t_i < t \leq t_{i+1}$, by linear interpolation we have

$$f_t(p) = \frac{t_{i+1} - t}{t_{i+1} - t_i} f_i(p) + \frac{t - t_i}{t_{i+1} - t_i} f_{i+1}(p).$$

145 Combined use of piecewise-linear approximations in both space and time
 146 provides an approximated model \tilde{F}_f of F_f that is continuous in the domain
 147 $D \times [t_0, t_k]$. Because of its piecewise-linear structure, it is possible to analyze the
 148 deep structure of \tilde{F}_f in an exact way, as we will show in the following.

Transition	Effect
$l_a l_b \rightarrow l_a l_b$	no change
$mr \rightarrow rm$	a minimum moves from a to b
$s1 r \rightarrow r s1$	a 1-saddle moves from a to b
$s2 r \rightarrow r s2$	a 2-saddle moves from a to b
$s2 r \rightarrow s1 s1$	a 2-saddle splits into two 1-saddles
$s1 s1 \rightarrow s2 r$	two 1-saddles merge into one 2-saddle
$s2 s1 \rightarrow s1 s2$	switch between a 2-saddle and a 1-saddle
$m s1 \rightarrow r r$	a minimum collapses with a 1-saddle
$m s2 \rightarrow r s1$	a minimum collapses with a 2-saddle, which becomes a 1-saddle
$r r \rightarrow m s1$	a minimum and a 1-saddle appear from two regular points
$r s1 \rightarrow m s2$	a minimum and a 2-saddle appear from a regular point and a 1-saddle

Table 1: Label transitions at a flipping edge (a, b) : a transition $l_a l_b \rightarrow l'_a l'_b$ means that a has label l_a before the flip and label l'_a after it, and similarly for b . Symmetric transitions and transitions involving maxima, which are analogous to those involving minima, are not listed for brevity.

149 **2.3 Tracking critical points**

150 Our approach to feature tracking is inspired to a mechanism first introduced
151 in [4] for controlled-topology filtering. We introduce the following classification
152 (labeling) for a node p of G_D :

- 153 – m: p is a minimum;
- 154 – M: p is a maximum;
- 155 – s1: p is a 1-saddle;
- 156 – s2: p is a 2-saddle;
- 157 – r: p is a regular point (neither of the above).

158 We are interested in events that change the classification of p during the diffusion
159 process (i.e., while varying t). From the piecewise-linear model introduced in
160 Section 2.1, it turns out that these events can occur only when two adjacent
161 samples flip the order of their values. Let a and b be two adjacent nodes on
162 the triangular tiling of D . We say that edge (a, b) *flips* at time t if and only if
163 $f_t(a) = f_t(b)$ while for an arbitrarily small $\varepsilon > 0$ we have either $f_{t-\varepsilon}(a) < f_{t-\varepsilon}(b)$
164 and $f_{t+\varepsilon}(a) > f_{t+\varepsilon}(b)$, or vice-versa. Let us consider two consecutive times t_i
165 and t_{i+1} in the discrete time sequence. Since \tilde{F}_f is linear between t_i and t_{i+1} ,
166 edge (a, b) flips in $[t_i, t_{i+1}]$ if and only if $f_{t_i}(a) < f_{t_i}(b)$ and $f_{t_{i+1}}(a) > f_{t_{i+1}}(b)$,
167 or vice-versa. The exact time t of flip is obtained by linear interpolation:

$$t = \frac{(f_{i+1}(a) - f_{i+1}(b))t_i + (f_i(b) - f_i(a))t_{i+1}}{f_i(b) - f_i(a) + f_{i+1}(a) - f_{i+1}(b)}.$$

168 We select all edges that flip in each interval $[t_i, t_{i+1}]$ and we sort them by
169 time of flip. This sequence of flips provides all events that may cause changes in

170 the deep structure. Given the flip of edge (a, b) at time t and given a labeling
 171 for a and b at time $t - \varepsilon$, the transitions described in Table 1 may occur.

172 We generate the deep structure of scale-space as follows. We evaluate all
 173 critical points of input signal f and we initialize a data structure listing them
 174 all. The data structure is an array of lists: each entry in the array is aimed at
 175 recording the trajectory of a critical point p through time, and it is linked from
 176 the corresponding entry of p in the grid G_D ; each entry in a list contains the
 177 location on G_D of the corresponding critical point at a given time t . For each
 178 2-saddle we generate two entries in the array, accounting for the fact that two
 179 1-saddles start at the same place and can proceed along the same trajectory
 180 until they possibly split, or one of them collapses.

181 For each $i = 0, \dots, k - 1$, we sort by time all flips that occur in $[t_i, t_{i+1}]$.
 182 Then we scan the sequence of flips, and we update the trajectories and the
 183 corresponding links from G_D according to relevant transitions (all except the
 184 first one in Table 1) that occur. A transition can be easily detected by comparing
 185 the height of a and b with all their neighbors just before and just after the time
 186 of split. Each transition that generates a pair of newborn critical points adds
 187 two new entries to the array of trajectories.

188 At the end of processing, the trajectory of each point consists of a chain of
 189 edges of the triangular tiling of D , with time stamps that record the “time of
 190 arrival” of the critical point at each node along the chain. For newborn critical
 191 points and for critical points that collapse we also record their pairing with the
 192 critical points that are born, or collapsed, together with them.

193 The array encoding trajectories will thus contain the lives of all critical points
 194 that appear in the deep structure, each encoded in a list. With a linear scan of
 195 the array, relevant points with respect to a given criterion can be extracted;
 196 the life span of each of them can be evaluated in constant time; while their
 197 position at any arbitrary time t can be evaluated with a binary search of the
 198 list and a linear interpolation. This mechanism supports several queries: critical
 199 points alive at a given scale t can be found without analyzing f_t , and they can
 200 be located either on f_t , or at their original position in f (or at time of birth
 201 for newborn points); the trajectory of each critical point can be obtained; most
 202 relevant points can be extracted based on the length of their life, and they can
 203 be located in space at any scale during their life span; given an interval of scales
 204 $[t, t']$, points whose life span contains the given interval can be extracted; etc.

205 3 Experimental results

206 We present results on two datasets: a 1200×1200 digital terrain model of a region
 207 around the Monte Rosa Massif in the Western Alps [3]; and a 500×500 range
 208 image from the Texas 3D Face Recognition Database [5]. For each dataset, we
 209 have built a sequence of snapshots of the scale-space on an exponential scale, by
 210 applying cumulative Gaussian blur, starting at $\sigma = \sqrt{2}$ and doubling the scale at
 211 each level, up to scale 1024 for terrain, and up to scale 128 for the range image.
 212 In Table 2 we show statistics about the deep structure of the terrain dataset.

scale	#Max (newborn)	#min (newborn)	#saddle (newborn)	avg life (newborn)	avg log life (newborn)	transition type	# events
3.0	1718(1164)	1270(1100)	2903(2201)	1.40(0.61)	10.34(2.32)	no change	1,100,422
9.0	1171(903)	906(863)	2062(1705)	1.34(0.58)	12.94(2.41)	move	13,490
27.0	766(626)	602(593)	1347(1168)	1.23(0.51)	16.48(2.46)	collapse	5,130
81.0	401(336)	285(285)	680(596)	1.25(0.48)	23.94(2.65)	birth	2,579
243.0	42(25)	18(18)	61(42)	2.86(0.69)	80.32(6.07)		

Table 2: Statistics on terrain dataset. Left: number of critical points and average duration of their lives in linear and logarithmic scale; in brackets statistics on newborn critical points. Right: transitions (flips) in the scale-space that: do not change the deep structure; move a critical point; collapse a pair of critical points; generate a pair of critical points.

213 Note that, thanks to our data structure, we query scales different from those of
 214 the discrete sequence in input. On the left side, we report the number of critical
 215 points found at each scale, together with their average duration of life through
 216 scale-space, both on a linear scale (which tends to give longer lives to features
 217 appearing at large scales) and on a logarithmic scale (which better spreads the
 218 life spans through scales that grow exponentially). We give statistics both on
 219 the total number of critical points, and on newborn critical points that were
 220 generated from birth transitions during the diffusion process.

221 Our exact tracking reveals a drawback of linear scale-space, which was un-
 222 derestimated in previous literature: a large number of newborn critical points
 223 are generated, a few of which live long and may sometimes take the place of
 224 original features. For instance, the maximum representing the highest point of
 225 Monte Rosa Massif at large scales is *not* a translation of the highest peak in
 226 the original dataset, but rather a new peak born around scale 32, while the true
 227 peak collapses short after. Substitution of features with newborn ones is hard to
 228 track, thus preventing a correct estimation of true life span in some cases. This
 229 is probably a good reason to prefer other kinds of scale-spaces, which guarantee
 230 non-creation of local extrema - one of the axioms of scale-space theory that is
 231 violated by linear scale-space in dimension larger than one. One possibility is for
 232 instance to incorporate controlled-topology filtering, as described in [4].

233 In figure 2 we show trajectories of points through the scale-space, superim-
 234 posed to original terrain. For each point, we draw its trajectory through its life
 235 span. Time is encoded by color, ranging from blue to green to red on a logarith-
 236 mic scale. Points do not move much until the very coarse scales. Only the few
 237 points that survive very long may undergo non-negligible displacement (reddish
 238 long trajectories) to adapt to the extreme smoothing of the signal. In Figure
 239 3 we show close-ups of trajectories on both datasets. Note how pairs of points
 240 move toward each other and eventually collapse at different times (different col-
 241 ors along the lines). The marker in cyan depicts a maximum that survives up to
 242 the largest scale, in spite of being very close to a saddle in the input.

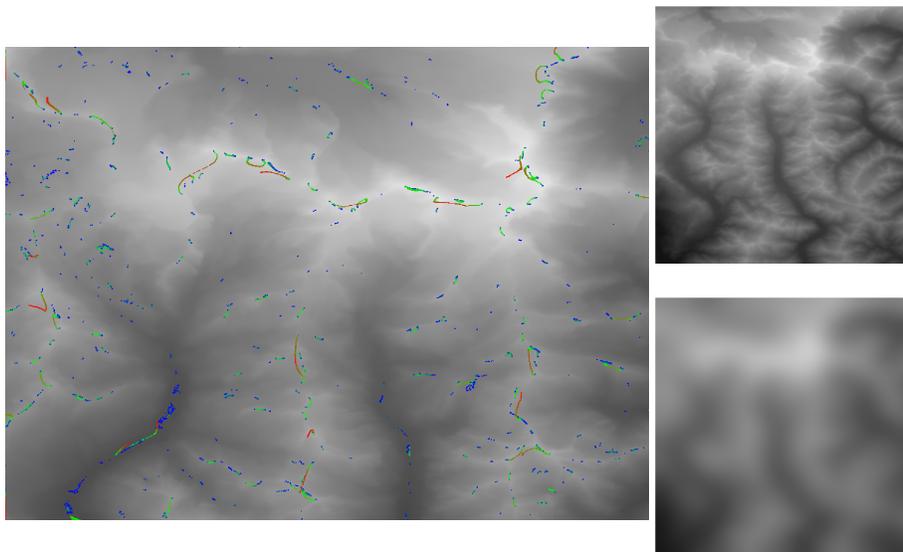


Fig. 2: Left: trajectories of all critical points on the terrain dataset (crop 900×700 pixels); color goes from blue to green to red, depending on the logarithm of time. Right: original terrain (top) and terrain filtered at scale 1024 (bottom). Zoom-in for a better view.

243 In Figure 4, we show results obtained on the range image, by querying the
 244 scale-space at scales 1 and 20 and by drawing critical points at their original
 245 positions in the input image (or at time of birth) with markers having a size
 246 proportional to the duration of life of each point. Note how several fiducial points
 247 that can be used for 3D face recognition [5] are detected by highly stable critical
 248 points. Since different fiducial points appear at different scales (e.g., centers of
 249 lips appear at a quite fine scale, while the center of the forehead appears at a
 250 very large scale), we are confident that combining information available from
 251 our scale-space model together with prior knowledge on distribution and proper
 252 scale of fiducial points, a reliable detection will be possible. We foresee that
 253 better results in this direction can be achieved by using a signal consisting of
 254 either Gaussian or total curvature of the underlying surface, rather than the raw
 255 range map. This will be the aim of our future work.

256 4 Concluding remarks

257 We have presented a framework that allows representing and querying the deep
 258 structure of scale-space for the critical points of scalar functions in a continuous
 259 setting. While we have described the framework for bi-variate functions defined
 260 on a rectangle and uniformly sampled at a grid, several extensions are possible.

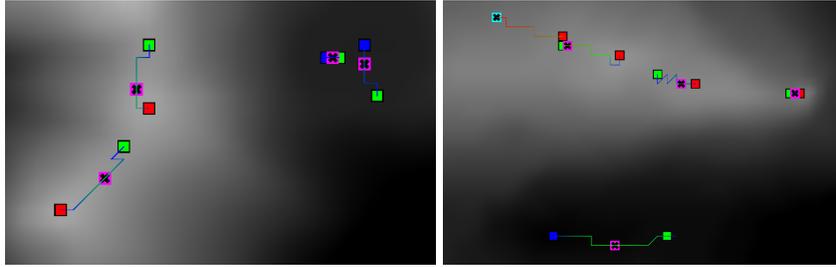


Fig. 3: Close-up of critical points trajectories on terrain (left) and on the range image (right). Markers: red maximum; blue minimum; green saddle; magenta position of collapse; cyan position of a surviving point at the largest scale.

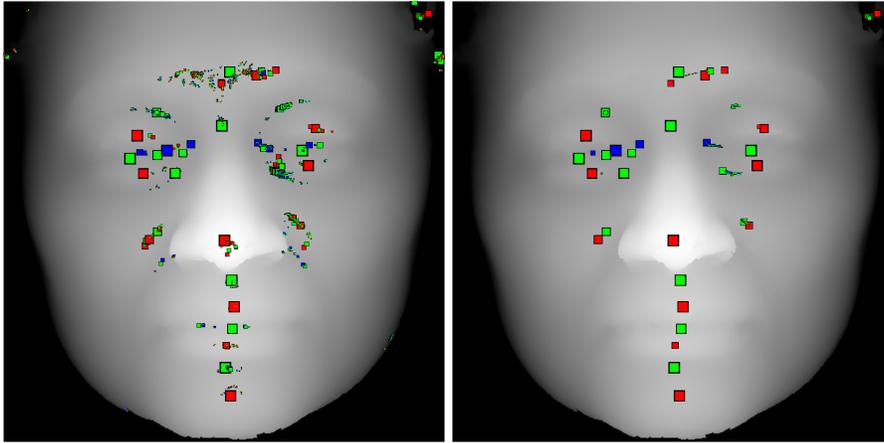


Fig. 4: Critical points alive at scale 1.0 (right) and 20.0 (left) on the range image, superimposed to the original dataset and depicted at time of their birth. Markers: blue minimum, red maximum, green saddle; size proportional to duration of life.

261 The framework is already modular with respect to the type of scale-space, which
262 affects just the construction of the input sequence of snapshots (f_0, \dots, f_k) .
263 Extension to irregularly sampled data, possibly on a 2-manifold domain (such
264 as the surface of a 3D object), is straightforward, provided that a triangle mesh
265 representing the domain is given. The only difference is that multiple saddles
266 with index $k > 2$ may appear. For dimensions higher than two, the mechanism
267 can be easily adapted, by taking into account the presence of different kinds of
268 saddles, according to Morse theory, and of related transitions.

269 The main limitation of the current method is that it can only track critical
270 points of a scalar field. Other features defined as zero-crossings of differential
271 properties could be also treated, but generalization is not always straightforward.
272 The essential ingredient to reproduce the framework for other kinds of features
273 is determining the events that may change the deep structure (like edge flips in

our case) and to sort them by time. Once these events are found, and related transitions are defined, the rest of the framework will work unchanged.

Our analysis reveals that linear scale-space may generate many newborn features, some of which may take the place of original features, thus hindering a correct feature tracking. For this reason, we plan to incorporate controlled-topology filtering [4] in our future work. Integration of this mechanism comes almost for free, being also based on the analysis of the same sequence of flips.

We plan to apply our framework to the analysis of terrains and to the extraction of fiducial points for 3D face recognition. Other extensions of the framework may involve computation of the Morse complex and its evolution through scale-space, and the evaluation of topological persistence, to better characterize stability and strength of local features [11].

Acknowledgments

We wish to thank Alan C. Bovik and the Lab for Image and Video Engineering of the University of Texas for kindly providing the 3D Face Recognition Database.

References

1. Bauer, D., Peikert, R.: Vortex tracking in scale-space. In: Proceedings of the symposium on Data Visualisation 2002. pp. 233–ff. VISSYM '02 (2002)
2. Correa, C., Ma, K.L.: Size-based transfer functions: A new volume exploration technique. *IEEE Transactions on Visualization and Computer Graphics* 14, 1380–1387 (2008)
3. de Ferranti, J.: Viewfinder panoramas, <http://www.viewfinderpanoramas.org/dem3.html>
4. Gingold, Y.I., Zorin, D.: Controlled-topology filtering. *Comput. Aided Des.* 39(8), 676–684 (Aug 2007)
5. Gupta, S., Castleman, K., Markey, M., Bovik, A.: Texas 3d face recognition database. In: *Image Analysis Interpretation (SSIAI)*, 2010 IEEE Southwest Symposium on. pp. 97–100 (2010)
6. Kindlmann, G.L., Estepar, R.S.J., Smith, S.M., Westin, C.F.: Sampling and visualizing creases with scale-space particles. *IEEE Transactions on Visualization and Computer Graphics* 15(6), 1415–1424 (Nov 2009)
7. Klein, T., Ertl, T.: Scale-space tracking of critical points in 3d vector fields. In: *Topology-based Methods in Visualization, Mathematics and Visualization*. pp. 35–50. Springer (2007)
8. Koenderink, J.J.: The structure of images. *Biological Cybernetics* 50, 363–370 (1984)
9. Lindeberg, T.: *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers (1994)
10. Price, K.: Annotated computer vision bibliography, <http://www.visionbib.com/bibliography/contents.html>
11. Reininghaus, J., Kotava, N., Guenther, D., Kasten, J., Hagen, H., Hotz, I.: A scale space based persistence measure for critical points in 2d scalar fields. *IEEE Transactions on Visualization and Computer Graphics* 17(12), 2045–2052 (2011)
12. Witkin, A.P.: Scale-space filtering. In: *Proc. 8th Int. Joint Conf. Art. Intell.* pp. 1019–1022. Karlsruhe, Germany (August 1983)