J. Vis. Commun. Image R. 30 (2015) 299–311

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

A topology-based approach to computing neighborhood-of-interest points using the Morse complex ☆

Ricardo Dutra da Silva [a,*], William Robson Schwartz [b], Helio Pedrini [c], Jesus Pulido [d], Bernd Hamann [d]

[a] Informatics Department, Federal University of Technology, Curitiba, PR 80230-901, Brazil
[b] Department of Computer Science, Universidade Federal de Minas Gerais, Belo Horizonte, MG 31270-010, Brazil
[c] Institute of Computing, University of Campinas, Campinas, SP 13083-852, Brazil
[d] Institute for Data Analysis and Visualization, Department of Computer Science, University of California, Davis, CA 95616-8562, USA

ARTICLE INFO

Article history:
Received 13 August 2014
Accepted 3 May 2015
Available online 22 May 2015

Keywords:
Interest points
Topological neighborhood
Matching
Correspondence
Morse complex
Local context
Image descriptors
Geometric similarity

ABSTRACT

A central problem in image processing and computer vision is the computation of corresponding interest points in a given set of images. Usually, interest points are considered as independent elements described by some local information. Due to the limitations of such an approach, many incorrect correspondences can be obtained. A specific contribution of this paper is the proposition of a topological operator, called Local Morse Context (LMC), computed over Morse complexes, introduced as a way of efficiently computing neighborhoods of interest points to explore the structural information in images. The LMC is used in the development of a matching algorithm, that helps reducing the number of incorrect matches, and obtaining a confidence measure of whether a correspondence is correct or incorrect. The approach is designed and tested for the correspondence of narrow-baseline synthetic and specially challenging underwater stereo pairs of images, for which traditional methods present difficulties for finding correct correspondences.

1. Introduction

The correspondence or matching of image interest points is a basic step in computer vision that is used to find corresponding locations in different images for tasks such as image stitching, image registration, scene reconstruction, object detection and recognition [3,27,31,34,37,47,53]. A well explored advantage of the correspondence of interest points is that it allows matching in the presence of occlusions and changes in scale and orientation [46]. Furthermore, more reliable matching of images can be computed [11] and might be used to compute denser correspondences [30]. However, the approach may fail depending on the image acquisition method, noise corruption, and transformations between images [38].

This paper investigates the correspondence of interest points as an auxiliary step for establishing a denser set of correspondences to further construct 3D models and merge point clouds produced from images acquired at different time steps (video frames). The considered images are challenging, such that we should avoid incorrect matches due to multiple regions with similar characteristics. Furthermore, the confidence of a match being correct should be measured.

The basic stages for corresponding interest points between images include: (1) interest point detection, (2) description, and (3) matching. After these steps, (4) filtration (as SIFT-ratio) and (5) model fitting (RANSAC) are commonly used. We define these steps as the basic framework.

Interest points are usually considered as independent elements described by some limited local information, basically, the appearance of patches of pixels surrounding the point location [46]. The limited local information clearly is not able to discriminate between interest points in some cases. The difficulty arises from patches related to two interest points not having enough dissimilarity between them. Examples of such cases occur in smooth regions, repeated patterns and symmetries. Therefore, images with different regions of high similarity often incur in problems of discrimination due to the limited information to compute descriptors.

Besides such difficulties, many other may occur. For example, for the underwater images considered in this paper, besides the morphological nature of the regions in the images, which may create highly similar structures, noise may be present due to particles in the water and the acquisition may produce distortions and illumination differences.

☆ This paper has been recommended for acceptance by "Yehoshua Zeevi".
* Corresponding author.
E-mail address: ricardodutr@gmail.com (R.D. da Silva).

http://dx.doi.org/10.1016/j.jvcir.2015.05.001
1047-3203/© 2015 Elsevier Inc. All rights reserved.

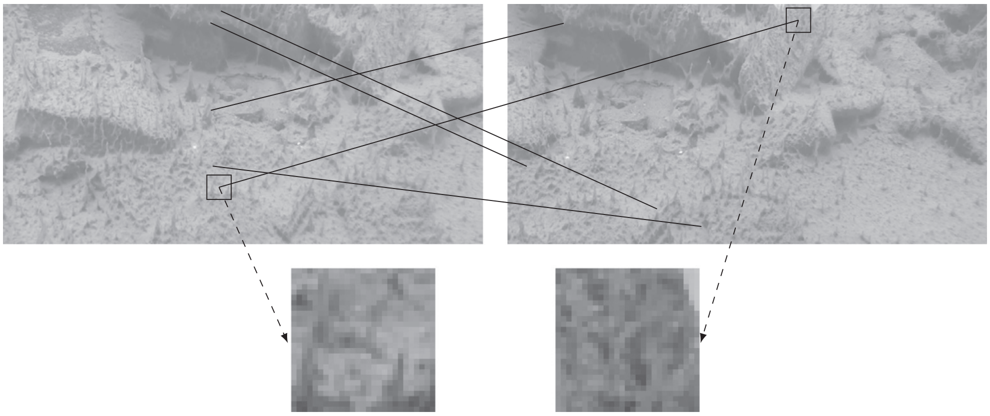


Fig. 1. Examples of incorrect matches computed due to regions of high similarity. The descriptor by itself is not able to discriminate between some regions. The zoomed regions show the pixel level texture similarities that produce close descriptors and consequently difficulties for the correspondence of interest points.

Commonly, the correspondences of images (step 3 of the framework) are computed by pairwise comparison of their interest point local descriptors, such as performed by *k*-nearest neighbor algorithm [34,46]. This approach is computationally efficient and suitable for real-time applications or for problems that deal with massive amounts of data. However, the difficulties arising from point detection and description are propagated and not properly handled at the matching stage. Fig. 1 shows some examples of incorrectly matched points.

The neighborhood relation proposed in this paper is mainly evaluated at the matching level. We use the neighborhood relation to produce local restrictions to the set over which correspondences are searched for. The idea is that, for the narrow-baseline images considered in this paper, spatially close interest points in one image should be related to spatially close putative correspondences in a second image.

Cases of incorrect correspondences are very common and there are some different levels in which it is possible to consider ways of eliminating them. At the early detection step, it is possible to threshold the detected interest points by using a measure of importance or of how salient the interest point is. In such a manner, it is possible to avoid detecting interest points in highly homogeneous regions.

At the matching level, approaches such as the one used in the Scale-Invariant Feature Transform (SIFT), the SIFT-ratio [32], perform a comparison of similarity between the *k*-closer matches of an interest point (usually *k* is 2) and, therefore, when they are too similar in terms of descriptors, the matches are removed. At another level, approaches such as RANdom SAmple Consensus (RANSAC) [17] use a model fitting to find a transformation such that outliers are removed from the correspondence set. It is clear that all these approaches reduce the number of correspondences. In fact, the initial number of correspondences invariantly suffers a drastic reduction. Fig. 2 shows an example of correspondences obtained through the application of interest point thresholding and SIFT-ratio. A set of good matches is acquired and can be useful in many applications, such as registration. However, the correspondences are usually very sparse.

A region growing approach is presented in [6] for correspondence verification. The method outperforms the selection of SIFT. However, it is still a method that produces sparse number of correspondences. A small number of correspondences may not be enough for obtaining denser matches for some images, such as the ones considered in this paper.

Alternatively, the structural relations between interest points can be introduced to obtain more global information for the matching process. Such relations are commonly modeled by using graphs and allow the exploration of structural arrangements to better discriminate regions in images. Successful applications to find or discriminate sparse number of interest points [7,22,25,51] have been reported. However, finding correspondences within dense sets of points using graphs is still a challenging problem. Algorithms for this purpose are computationally expensive and sensitive to noise in the data [28,40,52]. Much of the efforts to solve correspondences based on graphs consider such structures to connect all interest points [4,10,14,19,40,45], which makes the limitations of graph matching even more severe when the number of interest points is dense.

Computational topology is a field gaining importance for analyzing images at qualitative, structural and abstract levels [2,5,9,15,16,29,39]. In this work, we present an approach based on the topology of functions, given by Morse complexes, to defining locally meaningful connectivity of interest points. We have devised and tested the method for obtaining correspondences in images to demonstrate its contributions:

- A general neighborhood relation which can be adapted to different applications.

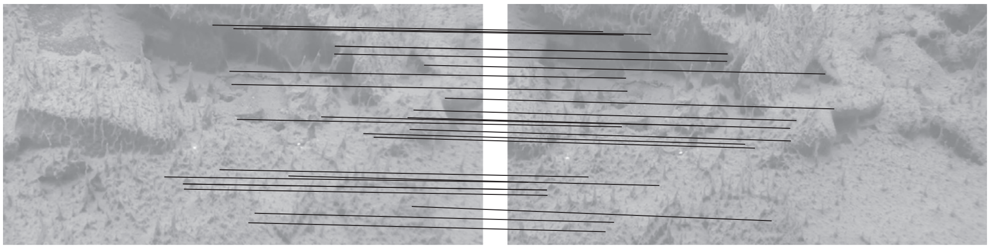


Fig. 2. Examples of output matches obtained through SIFT method. Incorrect correspondences are filtered out, however, the number of correspondences is drastically reduced and many regions do not contain any paired points.

- A manner of avoiding incorrect correspondences in high similarity images.
- An approach to evaluating the confidence of a correspondence being correct.

The matching algorithm developed is conceptually similar to pixel-based seed-growing methods such as [6,8,23,30] that have been proved useful. These are, however, mostly post-processing steps used after the matching of interest points. Our algorithm differs in its objective as it is used to compute more correct correspondences through an interest point-based growing. We essentially consider the three first steps of the basic image correspondence framework already mentioned. In the first step, we include the computation of the Morse complex and, consequently, the introduction of a neighborhood relation between interest points. This relation is used in the matching step (step 3 of the framework). In details, the steps are:

1.1. Detection of interest points: performed by identifying minimum and maximum Morse critical cells.
1.2. Topology computation: the connection between interest points is also obtained by using the Morse theory.
2. Description of interest points: this is performed through traditional descriptors such as HOG and SIFT.
3. Matching: the topological connections between interest points are used to produce a matching algorithm that restricts the search space and, by doing so, tries to avoid incorrect correspondences and produce a higher number of correct correspondences. In the current version of the method, rotation and scale transformations are not present in the tested datasets.

We use this framework to show that the topological information given by the Morse complex can be used to solve ambiguities produced by similar regions and augment the number of correct correspondences. The presented method is designed for the narrow-baseline images of our underwater application and it is shown to be effective. We believe that the method can be extended, in the future, to other types of images with wider baselines by generalizing the Morse complex to multiple scales and adapting the score measure presented in this paper.

The text is organized as follows. Section 2 presents basic concepts of Morse theory used in our approach. The proposed topological neighborhood is formulated in Section 3. In Section 4, the matching algorithm is developed. Experimental results are presented and discussed in Section 5. Section 6 concludes the paper with final remarks.

2. Background

This section reviews concepts required as basic foundation for the development of our neighborhood operator and matching algorithms. We present how images are modeled using cell complexes and how the discrete Morse complex is obtained from them. For a more complete understanding of the concepts one should refer to references in algebraic topology and computational topology [18,20,55].

2.1. Cell complexes

A p-cell is the building block to define a cell complex. The first three low-order p-cells are the 0-cells or nodes, 1-cells or edges and 2-cells or faces. Even though the secondary names are commonly used in computer science, we will avoid them. The concept of face is different in topology, as we will state soon. Therefore, to be consistent with the literature, we choose to use the primary names. Fig. 3 shows examples of p-cells up to dimension two. We suppress the superscript of a p-cell α^p, denoting it α, whenever the dimension is clear from the context.

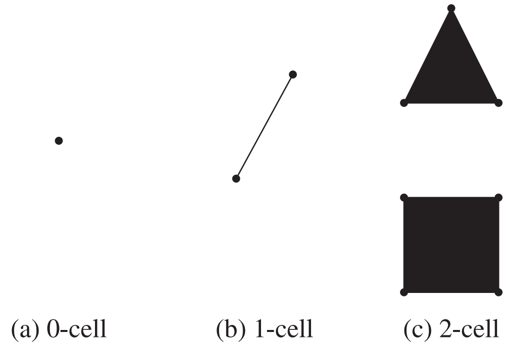

(a) 0-cell (b) 1-cell (c) 2-cell

Fig. 3. Examples of cells of dimension up to two. Figure adapted from [20].

The boundary of a p-cell consists of cells of dimension less than p that form the limit of the p-cell. The face of a p-cell σ^p is a cell τ^k, with $k \leqslant p$, which is part of the boundary of the p-cell. The p-cell σ^p is called a coface of τ^k. As such, we can say that the face of a cell bounds it. The bounding relations of face and coface will be stated as $\tau^p \preceq \sigma^k$ and $\sigma^p \succeq \tau^k$.

A cell complex K is a finite collection of cells such that all the faces of a cell in the complex also belong to the complex and the intersection of any two cell is either empty or a face of both cells. Fig. 4 shows examples of sets of cells that satisfy and that do not satisfy the conditions of cell complexes.

2.2. Images as cell complexes

Images are ordinarily a function $f : D \to \mathbb{R}$ defined on subset of the discrete lattice, $D = \{(x, y) \in \mathbb{Z}^2 | 1 \leqslant x \leqslant M, 1 \leqslant y \leqslant N\}$, such that a point p of D along with its value $f(p)$ is called a pixel. An image can be modeled by a regular 2-dimensional cell complex K, as arguments Kovalevsky [26]. Following the model of Robins et al. [41], the complex K has 0-cells related to the pixels over D. The 2-cells are squares/triangles defined by a pixel and some of its closest pixels in D. The 1-cells are faces of the 2-cells. An example of such a model is shown in Fig. 5.

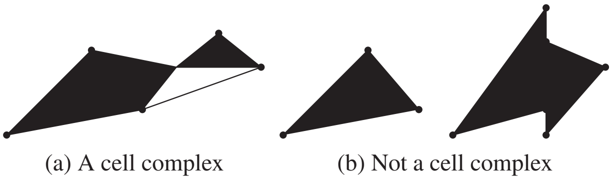

(a) A cell complex (b) Not a cell complex

Fig. 4. Sets of cell that satisfy (a) and that do not satisfy (b) the cell complex conditions. Figure adapted from [54].

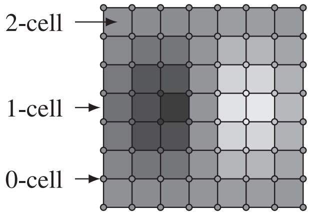


Fig. 5. Digital image modeled as a cell complex K. Pixels are 0-cells of the complex which also have cells of dimension 1 and 2 to explicitly define the topology of the image.

2.3. Discrete Morse theory

The discrete Morse theory relates the topology of a smooth function f with its critical points (maxima, minima and saddles), encoding all topological changes in the level sets of a function [18].

A function $f : K \to \mathbb{R}$ is a discrete Morse function if, for all $\alpha^p \in K, f$ takes a value less than or equal to $f(\alpha^p)$ in at most one coface of α^p and takes a value greater or equal to $f(\alpha^p)$ in at most one face of α^p,

$$\sharp\{\tau^{p+1} \succ \alpha^p | f(\tau^{p+1}) \leqslant f(\alpha^p)\} \leqslant 1, \quad (1)$$

$$\sharp\{\lambda^{p-1} \prec \alpha^p | f(\lambda^{p-1}) \geqslant f(\alpha^p)\} \leqslant 1, \quad (2)$$

where $\sharp$ denotes set cardinality. Given a discrete Morse function, a cell $\alpha^p \in K$ is critical if all cofaces take strictly greater values and all faces take strictly lower values in f,

$$\sharp\{\tau^{p+1} \succ \alpha^p | f(\tau^{p+1}) \leqslant f(\alpha^p)\} = 0, \quad (3)$$

$$\sharp\{\lambda^{p-1} \prec \alpha^p | f(\lambda^{p-1}) \geqslant f(\alpha^p)\} = 0. \quad (4)$$

In a two dimensional cell complex, critical 0-, 1- and 2-cells are, respectively, called minima, saddles and maxima. Fig. 6 shows the critical cells of an example cell complex.

A discrete Morse function on a complex K defines a discrete vector field V by pairing $\alpha^p \prec \beta^{p+1}$ whenever $f(\beta^{p+1}) \leqslant f(\alpha^p)$ [18] and producing a collection of pairs $\{\alpha^p \prec \beta^{p+1}\}$ such that each cell is in at most one pair. An example of Morse discrete vector field is shown in Fig. 7. The pairs of cells are depicted by drawing an arrow from a cell towards its coface in the pair.

Given a vector field V, the V-paths between critical cells can be used to compute the discrete Morse complex (DMC), M [41]. A V-path is a sequence of cells

$$\alpha_1^p, \beta_1^{p+1}, \alpha_2^p, \beta_2^{p+1}, \alpha_3^p, \ldots, \beta_{r-1}^{p+1}, \alpha_r^p \quad (5)$$

where $\alpha_j^p, \beta_j^{p+1} \in V, \beta_j \succ \alpha_{j+1}$, and $\alpha_j \neq \alpha_{j+1}$, for all $j = 1, \ldots, r-1$. Fig. 8 shows an example of a DMC where critical cells are connected by V-paths modeled using edges. Critical cells of maxima, minima and saddles are respectively depicted by the symbols $\odot, \circledcirc$ and $\oplus$.

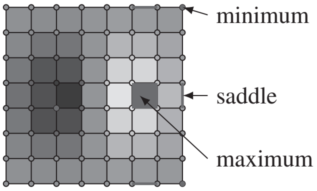


Fig. 6. Morse critical cells. Minima, saddles and maxima are respectively Morse critical cells of dimension 0, 1 and 2.

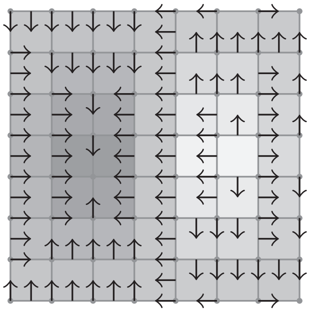


Fig. 7. Discrete Morse vector field with pairs of cells depicted by arrows.

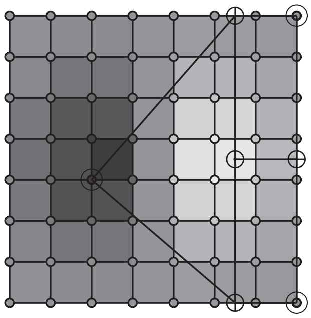


Fig. 8. Example of discrete Morse complex M.

Algorithms for computing the vector fields and the Morse complex are presented and analyzed in the paper of Robins et al. [41].

3. Local Morse Context

In this section, we define image interest points in terms of critical Morse cells and, mainly, introduce the Local Morse Context (LMC) operator to obtain the neighborhood relation of these interest points.

3.1. Morse complex and critical cells

As described in Section 2.3, minima and maxima are Morse critical cells. In computer vision, such features are usually obtained from a derivative of the input image function [35]. In the same manner, we use a derivative of the input image, the Laplacian of the Gaussian (LoG) [46], modeled as an image cell complex (see Section 2.2), to construct the discrete Morse complex (Section 2.3). Fig. 9 illustrates the process. Given the input image, the LoG of it is computed and used to obtain the DMC.

Critical cells of maxima and minima are analogous to the maxima and minima points usually detected as interest points of images. For that reason and to be in accordance with the proposed method, interest points will denote Morse critical cells and we will use both terms interchangeably in the remaining of the text. Critical Morse cells can also be saddles, however, they are not stable to perturbations in the input function and are not used as interest points. Therefore, given the DMC M, of an image, with n critical cells $\alpha_k^p, k = 1, \ldots, n$, we define the interest points to be the set of minima and maxima

$$C = \{\alpha_k^p | \alpha_k^p \in M \text{ with } p = \{0, 2\}, k = 1, \ldots, n\}. \quad (6)$$

Usually, an image interest point is a pixel for which the neighborhood relation is defined over its closest pixels in the discrete lattice (such as a 4- or 8-neighborhood). Such relations do not allow an easy determination of which interest points are close to each other. By using the discrete Morse complex, it is possible to define the neighborhood as a relation over closest interest points. The computation of such a relation is the topic of the next section.

3.2. Local Morse Context

The Local Morse Context (LMC) is a relation over the DMC for acquiring information regarding the neighborhood of interest points. Initially, some additional topological concepts are presented to support the definition of the LMC.

A subcomplex of a complex K is a subset of cells $L \subseteq K$ such that L is also a cell complex. The star of L contains all cofaces of $L, \mathrm{St}(L) = \{\alpha \in K | \alpha \succeq \tau \in L\}$. The closure of L is the smallest subset

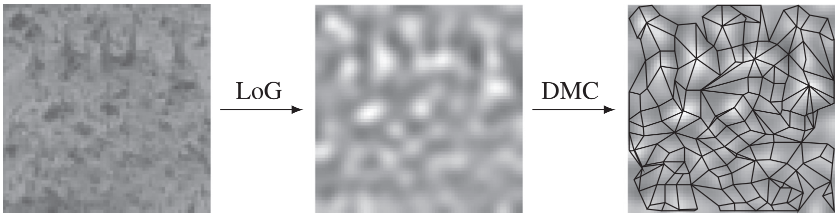


Fig. 9. Example of LoG and DMC of an image.

of K containing $L, \mathrm{Cl}(L) = \{\tau \in K | \tau \preceq \alpha \in L\}$. The link of L is the boundary of its star, $\mathrm{Lk}(L) = \mathrm{Cl}(\mathrm{St}(L)) - \mathrm{St}(\mathrm{Cl}(L) - \{\emptyset\})$ [55].

The main concepts of the proposed method are presented next. Given an interest point $\alpha_k \in C$ (see Section 3.1), the star of order i (or i-th iterated star) of $\alpha_k, \mathrm{St}_i(\alpha_k)$, is a recursion

$$\mathrm{St}_i(\alpha_k) = \begin{cases} \alpha_k & \text{if } i = 0 \\ \mathrm{St}(\alpha_k) & \text{if } i = 1 \\ \mathrm{St}(\mathrm{Cl}(\mathrm{St}_{i-1}(\alpha_k))) & \text{otherwise.} \end{cases} \tag{7}$$

The i-th order local Morse link (LML) of α_k is defined to be the link of the i-th iterated star of α_k,

$$\mathrm{LML}_i(\alpha_k) = \{\tau | \tau \in \mathrm{Lk}(\mathrm{St}_{i-1}(\alpha_k))\}. \tag{8}$$

The i-th order Local Morse Context of a critical cell α_k is defined as the set of minimum and maximum critical cells in the LMLs up to order i,

$$\mathrm{LMC}_i(\alpha_k) = \{\tau^p | \tau^p \in \mathrm{LML}_j(\alpha_k), p = 0, 2, j = 1 \ldots, i\}. \tag{9}$$

Fig. 10 illustrates the computation of the LMC_2 given a particular critical cell α. The second column shows the iterated stars of order 0 and 1, of α, while the third column presents the respective links, namely LML_1 and LML_2. The selection of maxima and minima is performed next, producing the sets shown in the fourth column. Finally, the union of the two sets of the fourth column produces the LMC_2, as can be seen in the rightmost image. Notice that the LMC is a nested operator, that is,

$$\mathrm{LMC}_0(\alpha_k) \subseteq \mathrm{LMC}_1(\alpha_k) \subseteq \cdots \subseteq \mathrm{LMC}_i(\alpha_k). \tag{10}$$

Fig. 10 shows the LMC_1 in the lighter region inside the darker rectangle related to the LMC_2.

The order of the LMC controls the extent of the neighborhood to be used. The higher is the order, the larger is the number of connected points and the region covered by the neighborhood. Therefore, more global information is captured as the order increases. Fig. 10 shows how the LMCs of order 1 (first row of fourth column) and 2 (fifth column), computed for a critical cell α, influence in the region covered around α.

The neighborhood given by the LMC makes it possible to explore the local structural information to characterize a pattern and/or increase the information of a local descriptor. Since the LMC is computed over a Morse complex, the neighborhood is based on the topology of a function and, therefore, it is expected to be independent of geometrical transformations that may be applied to the function.

4. Image matching using the LMC

This section presents how the LMC can be used to correspond points in pairs of images. Initially, we show a method for corresponding points within two LMCs, which is the core step for obtaining the correspondences between images. The correspondence of LMCs is used to guide the matching and, as a consequence, it helps to avoid incorrect matches due to similarities (as explained in the introduction).

4.1. Matching of LMCs

Given a critical cell α_k of a Morse complex M_1 and a critical cell β_l of a Morse complex M_2, let their LMCs be $\mathrm{LMC}_i(\alpha_k)$ and $\mathrm{LMC}_i(\beta_l)$, as illustrated in Fig. 11. Clearly α_k and β_l are corresponding points as can be noticed by the high similarity of the regions depicted in Fig. 11a and b. It can also be noticed that, as expected, the LMCs are also very similar, suggesting that the correspondence of LMCs can be formulated as the correspondence of structured data. The task becomes a graph matching problem which is well known and have various proposed solutions [10,19,50].

The standard solution of graph matching is carried out by means of graph isomorphisms or subgraph isomorphism if the graphs have different sizes. However, these methods can only find a solution if there is a perfect match [49]. In real world applications, finding isomorphisms is unfeasible because identical structures are very unlikely to occur due to noise and distortions present in the data.

There are many factors that may introduce noise and inaccuracy to image data. Such differences are probable to occur in the LMCs

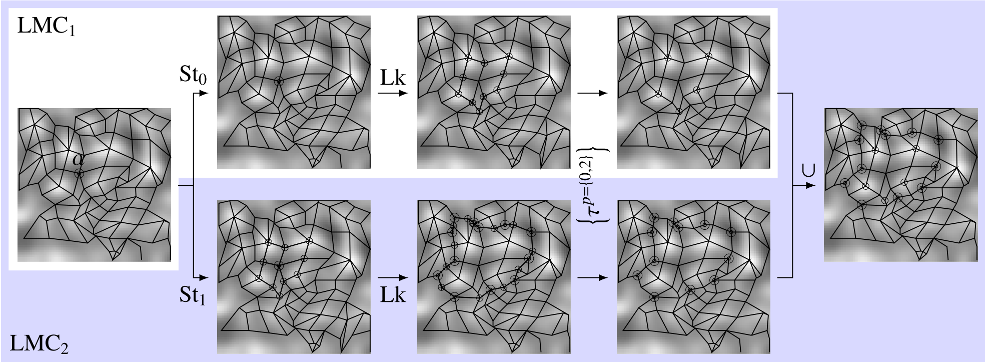


Fig. 10. Example of computing the LMC_2 of a maximum critical cell **α**. Increasing the order augments the number of critical points in the set and expanding the neighborhood.

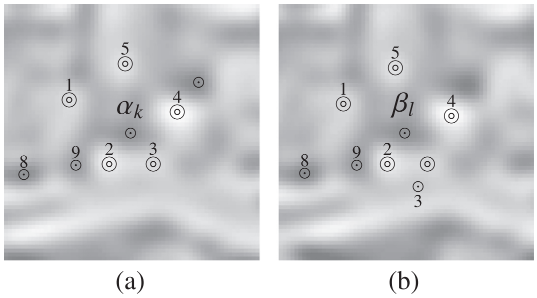


Fig. 11. Example of a pattern between two related critical cells. Missing or added critical cells are likely to occur. Correspondences are represented with numbers.

as well. Taking one of the LMCs as a base for comparison, some of its critical cells may be missing in the second LMC or some cells that do not appear in the first LMC could be added to the second LMC. In Fig. 11, one minimum point is missing in the second set and one maximum point appears in the second set, but not in the first.

Noisy or inconsistent cases are also a concern of various applications and extensively studied [4,10,19,40,44,48]. Usually, the matching constraints are relaxed in search of non-exact correspondences. The problem is known as inexact graph matching. We solve our problem with a method based on eigendecomposition approaches [43,45,50], specifically the one proposed by Scott and Longuet-Higgins [43], that provides an elegant solution with one-to-one correspondences and no explicit iterations.

Suppose $\mathrm{LMC}_i(\alpha_k)$ has m interest points, $\sigma_r, r = 1, \ldots, m$; and $\mathrm{LMC}_i(\beta_l)$ has n interest points $\tau_s, s = 1, \ldots, n$. Let $d_{rs} = \mathtt{dist}(\mathtt{desc}(\sigma_r), \mathtt{desc}(\tau_s))$ be the Euclidean distance between the descriptors ($\mathtt{desc}$) of σ_r and τ_s. The original method [43] uses distances between point coordinates, however, we use distances between descriptors, as proposed in [13], since this variation provided better results. Individually, each feature can be described by some local measure such as in HOG [12], HSC [42], SURF [1] or SIFT [31].

The method computes an $m \times n$ matrix G with pairwise affinities

$$G_{rs} = \exp\left(\frac{-d_{rs}^2}{2t^2}\right) \tag{11}$$

of interest points, where the parameter t controls the degree of proximity between descriptors and it is suggested to be 4 in [45]. Perfect matches ($d_{rs} = 0$) have affinity value 1. The farther the distance between descriptors, the more the affinity approaches 0.

A singular value decomposition (SVD) of G is performed

$$G = UDV^T \tag{12}$$

where U is an $m \times m$ orthogonal matrix, D is a $m \times n$ diagonal matrix and V^T is the transpose of an $n \times n$ orthogonal matrix V. Every element of the principal diagonal of D is replaced by 1 to create the matrix E. The association matrix is computed as

$$P = UEV^T \tag{13}$$

such that the rows of P index the interest points in $\mathrm{LMC}_i(\alpha_k)$ and the columns index the interest points in $\mathrm{LMC}_i(\beta_l)$.

If P_{rs} is the largest element both in row r and column s, then a strong correspondence will be achieved. However, if P_{rs} is the largest element in its column but not in its row, or, similarly, in row but not in column, then multiple points compete for the match and the correspondence is weak. Correspondences in the LMCs are obtained by retrieving the strong correspondences of P. The numbers in Fig. 11 show the correspondence between the two LMCs in our example.

4.2. Image match

Let C_I and C_J be the sets of interest points of two images I and J and also let S be a set with some correctly matched pairs (α_k, β_l), called seeds, such that $\alpha_k \in C_I$ and $\beta_l \in C_J$. Assume S is given (we show one way to compute it in Section 5). These sets are the inputs for Algorithm 1, which computes all the correspondences from C_I to C_J starting from the initial matches in S. The key idea is to increase the number of matchings at each iteration from matches already computed.

Algorithm 1. LMCImageMatching.

```
Input: Set of interest point $C_I$ from image $I$; set of interest
point $C_J$ from image $J$, set of seeds $S$.
Output: Putative matches.
1  $Q \leftarrow S$
2  while $Q$ is not empty do
3      extract match $(\alpha_k, \beta_l)$ from $Q$
4      compute correspondences $(\sigma_r, \tau_s)$ for
       $\sigma_r \in \mathrm{LMC}(\alpha)$ and $\tau_s \in \mathrm{LMC}(\beta)$
5      for each $(\sigma_r, \tau_s)$ do
6          if dist(desc($\sigma_r$), desc($\tau_s$)) <
           dist(desc($\sigma_r$), desc(pair($\sigma_r$))) then
7              if $(\sigma_r, \mathrm{pair}(\sigma_r)) \in Q$ then
8                  remove $(\sigma_r, \mathrm{pair}(\sigma_r))$ from $Q$
9              end
10             pair($\sigma_r$) $\leftarrow \tau_s$
11             $Q \leftarrow Q \cup (\sigma_r, \tau_s)$
12         end
13     end
14 end
```

An auxiliary queue Q is initialized in line 1 of the algorithm with the seeds in S. In the example of Fig. 12a, the set of seeds starts with three correspondences. Line 3 removes one putative match (α_k, β_l) from Q and line 4 computes the correspondences of the interest points in their LMCs (see matching of LMCs in Section 4.1). The correspondences in the LMCs provide new matches (σ_r, τ_s) in the neighborhoods of α_k and β_l. In the loop of line 5, the new matches are checked against possible previous matches of the same interest points. The loop goes through all (σ_r, τ_s) obtained checking whether the distance ($\mathtt{dist}$) between the descriptors ($\mathtt{desc}$) of σ_r and τ_s is closer than the descriptors of σ_r and a possible match previously found for it (line 6). The previous match is retrieved by the function pair. If the descriptors of σ_r and τ_s are closer, then pair(σ_r) will be set to τ_s and the putative match (σ_r, τ_s) will be inserted into the queue (lines 10 and 11). In such a way, new matches will be computed from the LMCs of σ and τ in a subsequent iteration of the outer loop of line 2. The condition in line 7 tests if a previous match of σ is in the queue and, if so, it is removed in line 8.

From the set of seeds, the number of correspondences grows until all interest points are matched, that is, until the queue Q becomes empty. The matching of LMCs, performed in line 4, enforces that interest points in a neighborhood of I are matched in the corresponding neighborhood of J. Therefore, the LMC guides the acquisition of new matches in Algorithm 1.

Fig. 12b–d depict different iterations in the growing process: initial, intermediary and final. The initial steps of Fig. 12b show

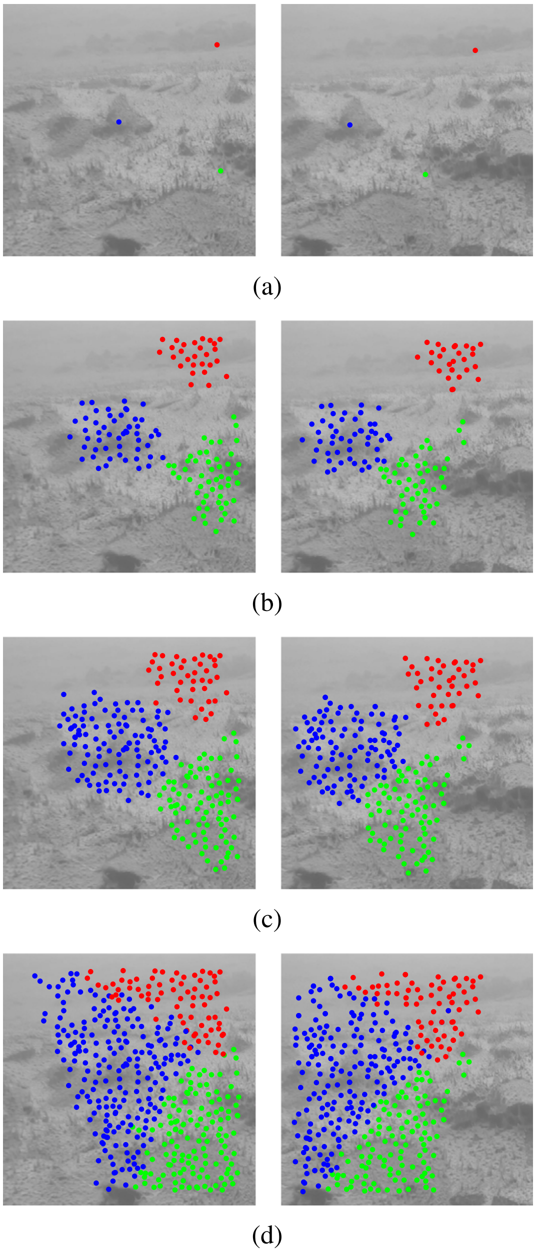


Fig. 12. Correspondences growing from initial seed set (a). The LMC locally guides the matches (b and c) until all interest points have correspondences established.

that the matches follow local restrictions. In such a manner, it is possible to avoid many incorrect matches as when using only the information of descriptors to perform the correspondences. Fig. 12c shows that the growing fronts from different seeds meet at some point. If the seeds were correctly initiated, and consequently the matches grew correctly, then the matches at the boundary of the fronts would agree and the growing process would stop due to the test of line 6. However, if the matches do not agree, one front will take over another front, correcting the previous matches. That can happen if one of the seeds is not a true match. In such a case, the matches will be corrected by the front of the correct seed. That means that the set S does not necessarily need to contain only correct matches, but at least one. When all the fronts meet and no better matches are found, the process stops as shown in Fig. 12d.

5. Experimental results

In this section, we present experimental results that show the effectiveness of the LMC to improve the matching of interest points and also to compute a score that characterizes correct and incorrect matches. The latter result is applied in the selection of seeds for Algorithm 1 and it is part of the setup of parameters for the algorithm.

5.1. Datasets and ground truth

We used three datasets to evaluate our method. The first dataset contains synthetic stereo pairs at three different baseline separations and ground truth disparity maps. The dataset is made available by the University of Alberta [36]. The second one is the Middlebury Stereo Datasets that also provides ground truth disparity maps [33]. The third dataset consists of pairs of underwater images taken from the bottom of lakes in Antarctica.

The first and second datasets are used as reference images since they have been used in the literature as benchmarks for stereo image pairs. These datasets also make available disparity maps that allow to directly evaluate the correctness of obtained matches. The synthetic images of the first dataset are built with different types of textures that contain repeated patterns and similar regions. The second dataset is composed of different types of real-world scenes.

The third dataset, due to several reasons, is more challenging in terms of matching interest points. These are non-calibrated images depicting ridged and peaked morphologies found on the bottom of lakes that create complex structural formations. In many cases, similar structures can be found all over the images, making it difficult for local descriptors to capture the differences between some regions. Besides the morphological nature of the regions in the images, many other difficulties arise from the acquisition of underwater images, such as distortions, illumination differences, and noise produced by particles in the water. A subset of images is shown in Fig. 13.

Unlike synthetic and Middlebury pairs, which have an available ground truth, the pairs of underwater images do not have a known ground truth characterizing matches in the images. Therefore, as in works that address a similar problem [34], we estimate a ground truth. Instead of a homography matrix, we use a fundamental matrix [21] to compute the ground truth due to the nature of the images in this work. Given a pair of images to be matched, a set of visually confirmed matches is chosen and used to compute the fundamental matrix. In such a way, it is possible to perform the rectification of the images and estimate corresponding regions between the two input images on the rectified image.

The tested datasets are composed of 90 pairs of synthetic images, 21 pairs of Middlebury images and 21 pairs of underwater images. Synthetic and underwater images were subdivided into subsets of validation images (9 synthetic pairs and 6 underwater pairs), used to set parameters; and of test images (81 synthetic and 16 underwater images) pairs, used to measure the quality of the matchings. The Morse complexes and interest points were obtained following the description in Section 3.1.

5.2. Evaluation metrics

In order to evaluate our results, we use metrics based on the number of correct and false matches, namely recall and 1-precision, since they are widely employed for similar evaluations [24,34].

Given the Morse interest points for all of the images in a dataset, two interest points α and β are considered a match if the

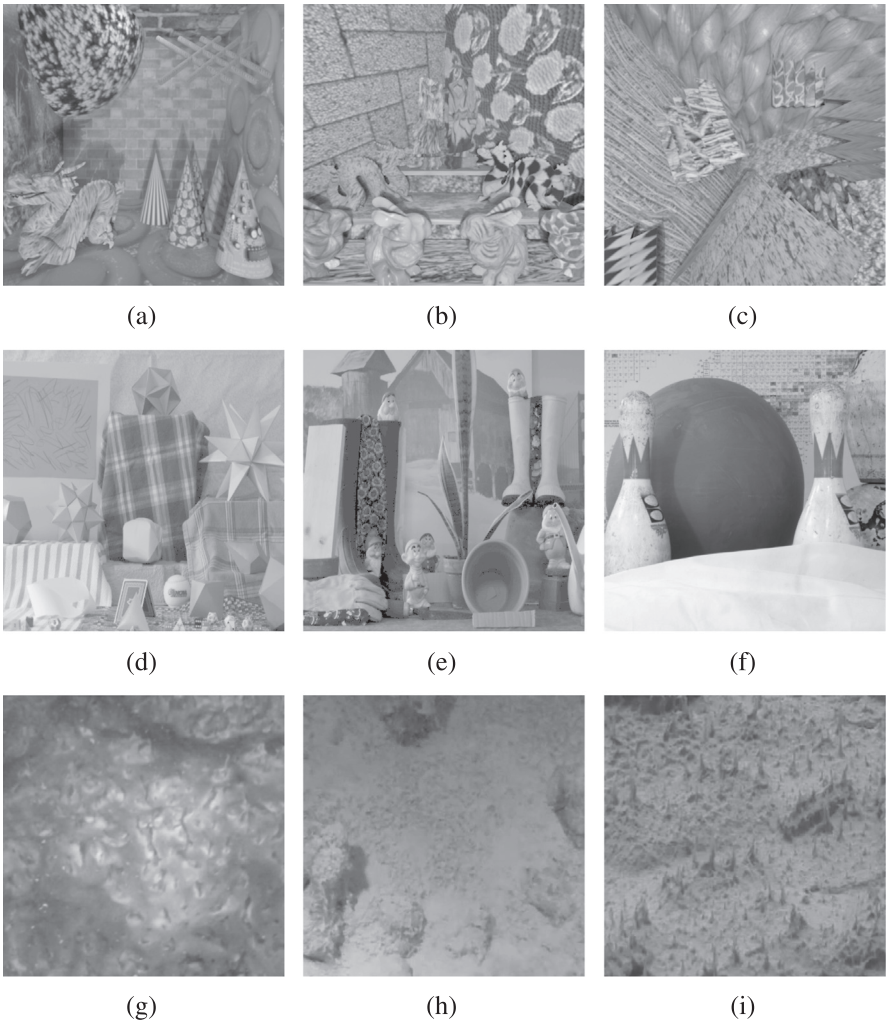


Fig. 13. Examples of images used in our experiments: (a)–(c) synthetic images from [36]; (d)–(f) Middlebury samples [33]; and (g)–(i) underwater images.

distance between their descriptors (Chi-squared distance in our experiments) is below a threshold t. A match is correct (true positive) if the interest points correspond to the same physical location (as determined by a ground truth), whereas a match is false (false positive) if the matched interest points correspond to different physical locations.

The correct correspondence of physical locations is determined by the overlap error [34]. Suppose A and B are the regions around α and β, respectively, for which the descriptors are computed. The overlap of A and B is defined by the ratio of the intersection and union of the regions $\epsilon_S = 1 - (A \cap TB)/(A \cup TB)$ under a transformation T. The transformation T is given by the disparity for the synthetic and Middlebury image pairs and by the rectification matrices for the underwater image pairs. As in [34], we assume that a match is correct if $\epsilon_S < 0.5$, so that the area covered by two corresponding regions is less than 50% of the region union.

Recall and 1-precision are defined as [24]:

$$\text{recall} = \frac{\text{number of true positives}}{\text{total number of positives}} \tag{14}$$

and

$$1 - \text{precision} = \frac{\text{number of false positives}}{\text{total number of matches (correct or false)}}. \tag{15}$$

The total number of positives for the given dataset is computed by comparing the overlap error of all interest points. The recall versus 1-precision graphs are obtained by varying the value of t.

5.3. Method setup

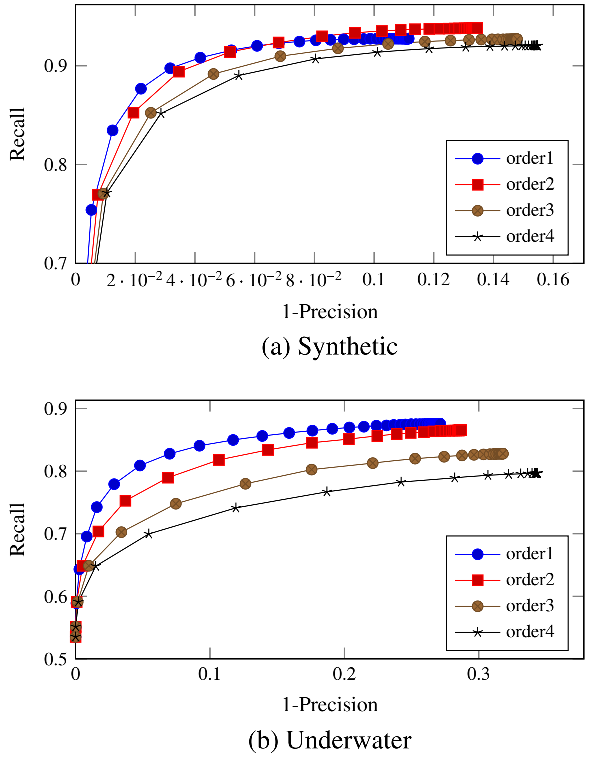
In this section, we discuss how to tune the following parameters of the matching algorithm: (i) order of the LMC used to grow matches; (ii) order of the LMC to score matches and select seeds; and (iii) number of seeds.

We used the histogram of oriented gradients (HOG) [12], a well known descriptor, to describe each interest point of the images. The descriptor is computed within a region of 17 × 17 pixels around each interest point. However, the proposed method is not attached to a specific descriptor. The Gaussian filter used for the LoG, as described in Section 3.1, has size 7 × 7 and $\sigma = 1.5$.

5.3.1. Order of LMC to grow matches

This test evaluates the choice of the LMC order to grow the number of matches in Algorithm 1. The order is chosen to maximize the confidence of the resulting matches. For these tests, the seeds for Algorithm 1 were visually chosen so that the algorithm grows from true matches.

The results for the four orders studied are shown in Fig. 14, both for synthetic and underwater datasets. The LMC of order 1 achieves the best results for this test. This behavior can be expected since



(a) Synthetic

(b) Underwater

Fig. 14. LMC order influence on growing the number of matches. The lower orders have a higher recall value. Therefore, the LMC-based algorithms performs better if the neighborhood considered to grow matches is smaller, the search for new correspondences in related regions between images is more restrict.

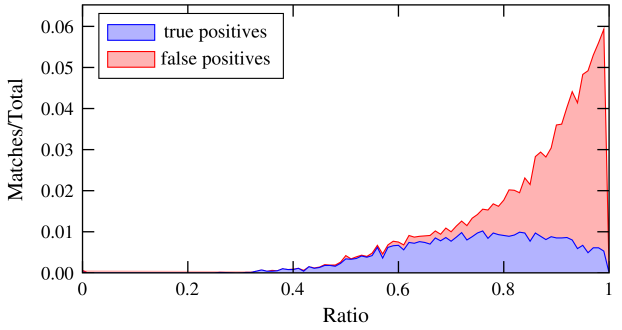


Fig. 15. SIFT-ratio of matches for underwater images. Many true positives have SIFT-ratio higher than 0.8.

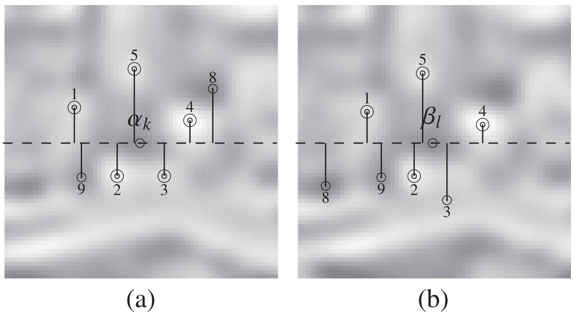


Fig. 16. Geometrical property for score where corresponding points in LMCs have similar distances to the horizontal lines.

the local restriction to grow is loosened as the order increases, allowing farther points to be matched. At lower orders, the local restriction is stronger, meaning that points that are close to one region of the base image should correspond to close points in the related region of the pair image. In this case, the LMC works as a neighborhood analogous to 4-pixel, 8-pixel or 16-pixel neighborhood of an image, except that in the LMC the neighbors are interest points instead of pixels.

5.3.2. Order of LMC to score matches

The second test we performed was to choose an order of the LMC to score matches and consequently estimate correct and incorrect matches. This is another application of the LMC we introduce and apply directly to choose seeds for our matching algorithm. Alternatively, SIFT-ratio could be a measure to choose seeds. However, in the experiments for our underwater test cases, it is possible to notice that there is a significant amount of true positives with SIFT-ratio higher than 0.8 (Fig. 15), also confirming the self-similarities of regions in the images.

Given two matched interest points $\alpha_k \in M_1$ and $\beta_l \in M_2$, we intend to estimate if they are a correct match. The LMCs of two matched points have relational information that can be used to increase the knowledge of how reliable their matching is.

If α_k and β_l are correctly matched, it is expected that their LMCs share similar structural relations, since the corresponding regions of the images are the same. Therefore, given a correspondence of the points in their LMCs, it is possible to obtain a measure of how similar the two patterns are. We propose to compute a matching score as

$$\mathrm{score}(\lambda_r, \tau_s) = \sharp\{(\lambda_r, \tau_s) | Pr(\lambda_r) \approx Pr(\tau_s)\} \tag{16}$$

where (λ_r, τ_s) are corresponding cells between LMCs that have values $Pr(\lambda_r)$ and $Pr(\tau_s)$ for some property on the structural patterns. The symbol $\sharp$ denotes set cardinality so that the score counts the number of corresponding cells that agree with respect to property *Pr*. The property can be some invariant characteristic shared by the images in one application. Notice that the range of score values is dependent on the number of points in the LMCs.

Taking Fig. 16 as an example, we need to find a property that holds for the pairs of images from our dataset. The images considered in this work present perspective transformations which do not preserve angles and ratios of lines linking interest points. Let the central circles be a matched pair of points (α_k, β_l) to be scored. The other circles are interest points in the LMCs such that the numbers define their correspondences. Consider also the horizontal lines based on α_k and β_l. Due to the nature of the stereo pairs, the points in the contexts may have significant horizontal displacements relative to the central point. However, the distances from corresponding points in the LMCs to the horizontal lines are expected to be similar (see vertical, solid lines in Fig. 16).

We use this fact to score matches. Given a match pair (α_k, β_l), we define its score as

$$\mathrm{score}(\alpha_k, \beta_l) = \sharp\{(\sigma_r, \tau_s) | \mathrm{disp}(\alpha_k, \sigma_r) - \mathrm{disp}(\beta_l, \tau_s) \leqslant c\} \tag{17}$$

where $\mathtt{disp}(\alpha_k, \sigma_r)$ $(\mathtt{disp}(\beta_l, \tau_s))$ is the vertical difference from the points σ_r (τ_s) in the LMC of α_k (β_l) to the horizontal lines of α_k (β_l). The difference between displacements is signed to differentiate between points lying below and above the horizontal lines. The constant c controls how much the vertical distance can differ between corresponding points. For our images, we have found that $c = 3$ leads to satisfactory results. In our example, except from interest points identified with values 3 and 8, the displacements of five interest points are approximately the same. Therefore, the score for the matching (α_k, β_l) equals 5 and it suggests that the matching is probably a correct one. This is the behavior studied in the following tests.

Table 1
Best scores for each order of LMC. For each order of LMC, the best score was chosen, that is, a correspondence scored with this or a greater value is mostly probable to be correct (fifth column) and the probability of finding such a correspondence is also high (fourth column). Higher orders perform better in this case. We find out that an order of 3 and score 11 are good choices for measuring the confidence of correspondences.

Dataset	Order	Score	Number of matches	True positives
			(from total of matches)	
Synthetic	1	2	0.81	0.94
	2	5	0.84	0.95
	3	9	0.85	0.96
	4	13	0.85	0.96
Underwater	1	2	0.43	0.75
	2	7	0.55	0.90
	3	11	0.59	0.91
	4	15	0.59	0.90

The validation images were matched by using a 1-NN algorithm, which finds the closest correspondences in terms of their distance in the feature space, and all pairs of matched points were scored using orders 1–4 for the LMC. An adequate choice of the score is achieved when the number of correct matches becomes significantly larger than the number of incorrect matches and also the number of correct matches is not too small.

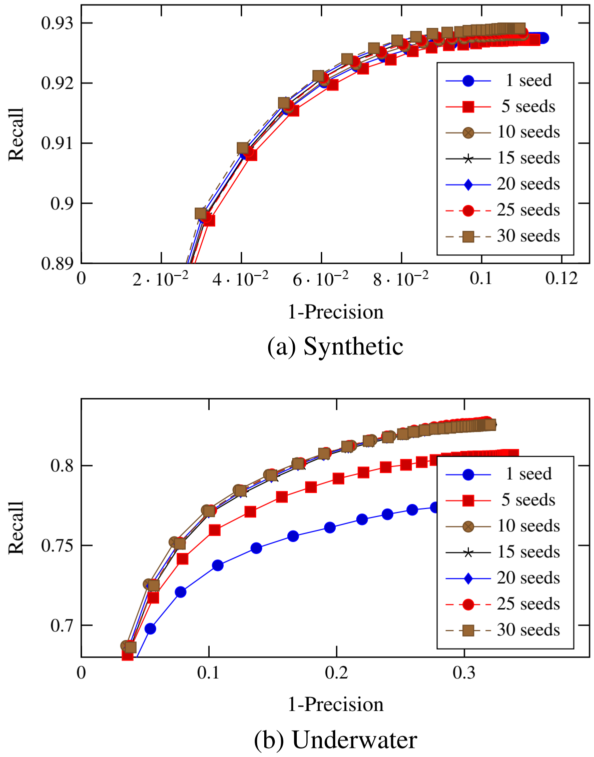
Therefore, for each order we chose the score which returned a considerable amount of matches with great probability of being a correct one. The results are summarized in Table 1, which shows the best score (column 3) for each order of LMC (column 2) for the two datasets. Column 3 shows the number of matches from the total expected to have a score equal to or greater than the chosen score, and column 4 shows the probability of having a correct match given that the score is equal to or greater than the chosen score.

The scores become more discriminative as the order increases. Orders 3 and 4 are highly discriminative, however, the results are very similar both for the number of matches and the probability of a good match. We use order 3 to compute scores, since it achieves similar behavior when compared to order 4 with less points in the LMC. The chosen score was 11 since the number of correct matches is substantially superior than the incorrect matches and it is a good score for both synthetic and underwater cases (from the score of 9 of the synthetic images, there is only a small reduction in the number of matches to 0.84).

5.3.3. Number of seeds

We use the previous results to compute seeds for Algorithm 1. We intend to choose k seeds such that the resulting matches are optimized. Particularly, the k seeds should be mostly correct seeds, so they were randomly chosen from matches agreeing with our previous choice of order 3 and score 11. The influence of number k of seeds is shown in Fig. 17. The matching results become better as the number of seeds increases, however, the gain practically stabilizes after 10 seeds. Therefore, approximately 10 seeds suffice for the images in our dataset.

It is still possible that incorrect seeds are chosen through the fixed parameters. We conducted an experiment to evaluate the influence of incorrect seeds among 10 selected seeds. In order to do that, subsets of 10 seeds were divided between k_c correct seeds and k_i incorrect seeds. The correct seeds were estimated as in the last experiment and the incorrect seeds were obtained by randomly choosing interest points in the two images and forming a seed with them. At least one correct correspondence is needed, therefore, we varied the number of correct seeds from $k_c = 1, 2, \ldots, 10$ and, consequently, the number of incorrect seeds in a set of 10 seeds is given as $k_i = 10 - k_c$. The experiment was performed with the underwater dataset since this is the one with the largest variation for the seeds. We show results in Fig. 18, where the cardinality of the subsets of 10 images is discriminated as $(k_c : k_i)$. The result of one correct seed in ten, $(1 : 9)$, achieves similar result as when choosing only one seed, as in Fig. 17b. The other results fluctuate between the previous result and when 10 seeds are correct. This suggests that the method performs well even when incorrect seeds are present in the input set for the matching algorithm.



(a) Synthetic

(b) Underwater

Fig. 17. Relation between the number of seeds and number of matches. The number of correctly corresponded points increases with the number of randomly chosen seeds, but converges near 10 seeds.

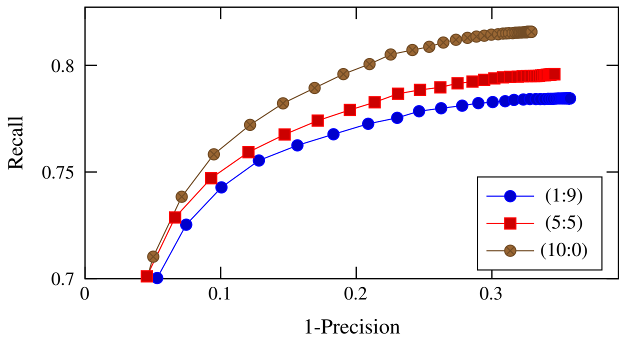
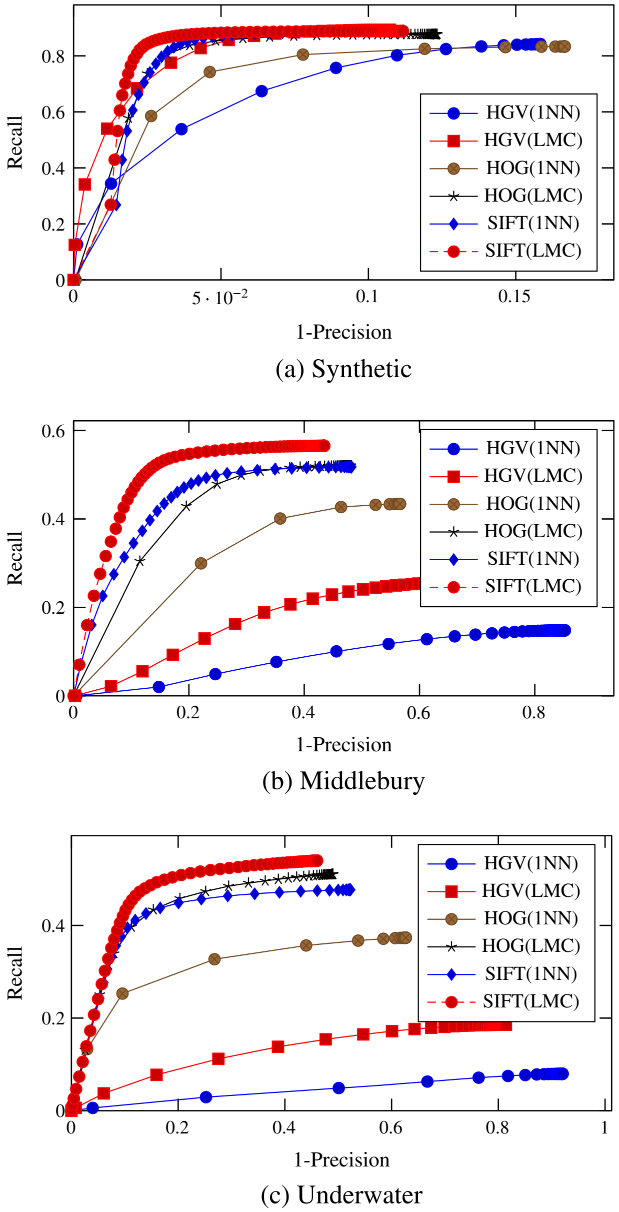


Fig. 18. Insertion of incorrect seeds in the algorithm. The number of incorrect seeds was arbitrarily fixed in 10 seeds. In the chart key, $(k_c : k_i)$ represents the choice of k_c correct matches and k_i incorrect matches. The result fluctuates according to the number of incorrect matches, however, it is bounded with similar results when using only one seed and ten seeds, as in Fig. 17b. This suggests the methods performs well even when incorrect seeds are in the input set for the algorithm.

5.4. Matching results

In this section, we show that our structural approach is more robust to find matches than an approach without structural



(a) Synthetic

(b) Middlebury

(c) Underwater

Fig. 19. Average results for all the test images using HGV, HOG and SIFT descriptors. The recall of the LMC-based matching is higher, independent of the descriptor, when compared to the results obtained using the 1-NN matching. The conflicts of descriptors are broken with the LMC neighborhood restriction, allowing to obtain more correct correspondences.

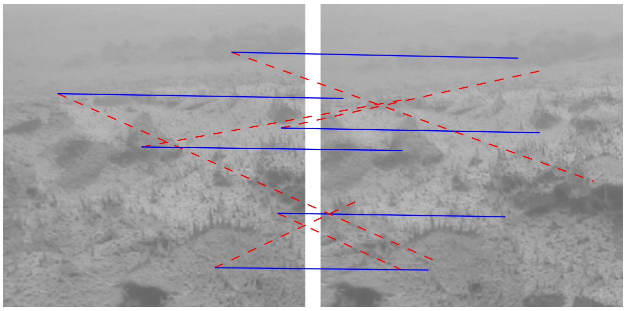


Fig. 20. Examples of interest points correctly corresponded due to the use of the LMC (solid lines) but incorrectly corresponded by using the nearest neighbor approach (dashed lines).

information (1-NN). To select the seeds for our method, we randomly choose an interest point in one image and match it in the second image using the 1-NN approach. The score for the match is computed by using LMC of order 3 and, if it is greater than or equal to 11, the match will be kept as a seed. The process is repeated until a set of 20 seeds (ideal number from the previous section) is acquired. The seeds computed in such a manner are the inputs for Algorithm 1.

The graphs in Fig. 19 show the average results for all the test images. The nearest neighbor and LMC matching algorithms are performed using three different descriptors: a simple histogram of gray values (referred to as HGV), HOG descriptor and SIFT descriptor. The first two descriptors are computed within a region of 17 × 17 pixels and the latter within a region of 16 × 16 pixels. The HOG is normalized with the L-norm, whereas SIFT uses a two-step illumination normalization. We show that the LMC-guided matching is able to improve the results for weak descriptors (HGV) and strong descriptors (HOG, SIFT). When using the LMC-based match, the respective curve for a given descriptor is pushed towards the top-left corner to a greater degree when compared to the nearest neighbor matching and the same descriptor.

The HGV is considered to obtain a baseline for the results. Since the descriptor is weak, many incorrect correspondences are expected, specially for the challenging images such as the ones from the underwater dataset. The lines of the matching driven by the LMC show the improvement in correct matches when compared to the nearest neighbor approach.

The HOG and SIFT descriptors, as expected, improve the results for both methods (ours and nearest neighbor) when compared to the HGV. However, the LMC method is still able to obtain further improvement for the matches. Such a behavior reflects the properties of the structure-based matching carried out by our method. The number of correct matches increases since conflicts of descriptors (interest points in different regions of an image but with close descriptors) are avoided with the neighborhood restriction to grow matches. Such behavior can be noticed in Fig. 20, where the solid lines show examples of matches that were correctly computed using the LMC-based algorithm while incorrectly corresponded with 1-NN.

The results show that LMC is a neighborhood relation that can be used to support image processing tasks. The tested correspondence of the images shows a particular case in which exploring the LMC helps traditional approaches. The proposed correspondence algorithm is limited to the types of images and transformations present in the experimented synthetic and underwater datasets. Rotations between images could also be evaluated with the current method since the matching between LMCs is not dependent on the image type and the score could be adapted to consider the direction of the central points in LMCs to find the horizontal base line. Scale transformations is still a challenge that would require further investigation on computing multiscale Morse complexes and neighborhood relations.

The complexity of the correspondence algorithm can be divided into three steps: the construction of a heap for the 1-NN used to find seeds, the process of actually finding seeds, and the matching given by Algorithm 1. The heap construction can be done in linear time on the number of interest points. The results on score and seeds suggest that we have more than a 50% chance of finding a seed at each random run of the 1-NN. This expectation has been confirmed by our experiments and, therefore, for 10 seeds we expect the process to run in $O(k \log n)$ with k approximately 20 runs of the 1-NN search and n features in an LMC. Finally, the computation of correspondences empirically suggests a linear time algorithm for small orders of the LMC. As the order increases, however, the constant multiplying the linear function can produce drastic augments on the computational time. The method is bounded by the SVD algorithm and, therefore, would take $O(MN^2)$ time in worst case if we had LMCs with all the interest points in the images, M points in the first image and N points in the second. The computation of the Morse complex can be performed in linear time on the number of cells in the 2-dimensional cell complex, as discussed in Robins et al. [41]. The algorithm has as input a filtration of the complex which involves a sorting of the 0-cells. This could bound the algorithm in $O(m \log m)$ for a complex with m 0-cells, however, alternatively, if the range of pixel values is properly limited, a linear sorting algorithm can be used, maintaining the linear complexity.

6. Conclusions

We have presented a topological operator, the Local Morse Context, to obtain neighborhoods of interest points. The LMC has been applied to find correspondences between interest points of stereo image pairs and also to compute a measure to quantify the confidence of matched points. This measure, denominated score, is effective for selecting matched pairs as seeds from which the number of matches is grown.

The matching algorithm explores the LMC neighborhood to produce correspondences in images agreeing with local proximity restrictions. As a consequence, the use of LMC avoids incorrect matches when the limitations of local descriptors do not allow a discrimination between various interest points. Finally, the LMC makes it possible to explore the topological relations between interest points in a general way that can be used for different types of images and applications.

The presented discrete Morse complexes are computed at a specific scale. We intend to study how to compute and model the Morse complexes such that the neighborhood can be generalized to multiple scales and applied to the correspondence of images under scale transformations. The persistent homology provides means of measuring the importance of topological features and it can be used to exploit such hierarchies of complexes.

The presented score measure is dependent on the definition of a geometrical property shared by the type of images under investigation. It would be interesting to define a more general score measure. There are some graph matching techniques based on edition distances that could be experimented to measure similarities between LMCs.

Acknowledgments

The authors are grateful to FAPESP, FAPEMIG, CNPq, and CAPES for the financial support. The underwater images were cordially by colleagues from the Department of Computer Science and Department of Earth and Planetary Sciences, University of California, Davis.

References

[1] H. Bay, T. Tuytelaars, L.J.V. Gool, SURF: speeded up robust features, in: European Conference on Computer Vision, Springer, Graz, Austria, 2006, pp. 404–417.
[2] P.T. Bremer, E.M. Bringa, M.A. Duchaineau, A.G. Gyulassy, D. Laney, A. Mascarenhas, V. Pascucci, Topological feature extraction and tracking, J. Phys: Conf. Ser. 78 (1) (2007) 1–5.
[3] M. Brown, R.I. Hartley, D. Nistér, Minimal solutions for panoramic stitching, in: IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 2007, pp. 1–8.
[4] H. Bunke, Error-tolerant graph matching: a formal framework and algorithms, in: International Workshop on Advances in Pattern Recognition, Springer-Verlag, London, UK, 1998, pp. 1–14.
[5] G. Carlsson, A. Zomorodian, A. Collins, L. Guibas, Persistence barcodes for shapes, in: Eurographics Symposium on Geometry Processing, ACM, New York, NY, USA, 2004, pp. 124–135.
[6] J. Cech, J. Matas, M. Perdoch, Efficient sequential correspondence selection by cosegmentation, IEEE Trans. Pattern Anal. Mach. Intell. 32 (9) (2010) 1568–1581.
[7] M. Chertok, Y. Keller, Efficient high order matching, IEEE Trans. Pattern Anal. Mach. Intell. 32 (12) (2010) 2205–2215.
[8] M. Cho, K.M. Lee, Bilateral symmetry detection via symmetry-growing, in: British Machine Vision Conference, BMVA Press, London, UK, 2009, pp. 4.1–4.11.
[9] A. Collins, A. Zomorodian, G. Carlsson, L.J. Guibas, A barcode shape descriptor for curve point cloud data, Comput. Graph. 28 (6) (2004) 881–894.
[10] D. Conte, P. Foggia, C. Sansone, M. Vento, Thirty years of graph matching in pattern recognition, Int. J. Pattern Recogn. Artif. Intell. 18 (3) (2004) 265–298.
[11] B. Cyganek, An Introduction to 3D Computer Vision Techniques and Algorithms, John Wiley & Sons, 2007.
[12] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 2005, pp. 886–893.
[13] E. Delponte, F. Isgrò, F. Odone, A. Verri, Svd-matching using sift features, Graph. Models 68 (5) (2006) 415–431.
[14] O. Duchenne, F. Bach, I.S. Kweon, J. Ponce, A tensor-based algorithm for high-order graph matching, IEEE Trans. Pattern Anal. Mach. Intell. 33 (12) (2011) 2383–2395.
[15] H. Edelsbrunner, J. Harer, A. Zomorodian, Hierarchical Morse complexes for piecewise linear 2-manifolds, in: Symposium on Computational Geometry, ACM, New York, NY, USA, 2001, pp. 70–79.
[16] B.D. Fabio, C. Landi, Persistent homology and partial similarity of shapes, Pattern Recogn. Lett. 33 (11) (2012) 1445–1450.
[17] M.A. Fischler, R.C. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Commun. ACM 24 (6) (1981) 381–395.
[18] R. Forman, Morse theory for cell complexes, Adv. Math. 134 (1) (1998) 90–145.
[19] X. Gao, B. Xiao, D. Tao, X. Li, A survey of graph edit distance, Pattern Anal. Appl. 13 (1) (2010) 113–129.
[20] L. Grady, J.R. Polimeni, Discrete Calculus – Applied Analysis on Graphs for Computational Science, Springer, 2010.
[21] R. Hartley, A. Zisserman, Multiple View Geometry in Computer Vision, second ed., Cambridge University Press, New York, NY, USA, 2003.
[22] H. Jiang, T.P. Tian, S. Sclaroff, Scale and rotation invariant matching using linearly augmented trees, in: IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 2011, pp. 2473–2480.
[23] J. Kannala, E. Rahtu, S. Brandt, J. Heikkila, Object recognition and segmentation by non-rigid quasi-dense matching, in: IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 2008, pp. 1–8.
[24] Y. Ke, R. Sukthankar, PCA-SIFT: a more distinctive representation for local image descriptors, in: IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, 2004, pp. 506–513.
[25] G. Kim, C. Faloutsos, M. Hebert, Unsupervised modeling and recognition of object categories with combination of visual contents and geometric similarity links, in: ACM International Conference on Multimedia Information Retrieval, Vancouver, Canada, 2008, pp. 419–426.
[26] V.A. Kovalevsky, Finite topology as applied to image analysis, Comput. Vision Graph. Image Process. 46 (2) (1989) 141–161.
[27] C.H. Lampert, M.B. Blaschko, T. Hofmann, Efficient subwindow search: a branch and bound framework for object localization, IEEE Trans. Pattern Anal. Mach. Intell. 31 (12) (2009) 2129–2142.
[28] M. Leordeanu, M. Hebert, A spectral technique for correspondence problems using pairwise constraints, in: IEEE International Conference on Computer Vision, IEEE Computer Society, Beijing, China, 2005, pp. 1482–1489.
[29] D. Letscher, J. Fritts, Image segmentation using topological persistence, in: Conference on Computer Analysis of Images and Patterns, Springer, Vienna, Austria, 2007, pp. 587–595.
[30] M. Lhuillier, L. Quan, A quasi-dense approach to surface reconstruction from uncalibrated images, IEEE Trans. Pattern Anal. Mach. Intell. 27 (3) (2005) 418–433.
[31] D.G. Lowe, Object recognition from local scale-invariant features, in: IEEE International Conference on Computer Vision, IEEE Computer Society, Washington, DC, USA, 1999, pp. 1150–1157.
[32] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vision 60 (2) (2004) 91–110.
[33] Middlebury, The Middlebury Stereo Datasets, 2015. <http://vision.middlebury.edu/stereo/data/>.
[34] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE Trans. Pattern Anal. Mach. Intell. 27 (10) (2005) 1615–1630.
[35] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, L.V. Gool, A comparison of affine region detectors, Int. J. Comput. Vision 65 (1-2) (2005) 43–72.
[36] D. Neilson, Y.H. Yang, Evaluation of constructable match cost measures for stereo correspondence using cluster ranking, in: IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 2008, pp. 1–8. <http://webdocs.cs.ualberta.ca/~stereo/datasets/>.
[37] E. Nowak, F. Jurie, B. Triggs, Sampling strategies for bag-of-features image classification, in: European Conference on Computer Vision, Springer Berlin Heidelberg, Graz, Austria, 2006, pp. 490–503.
[38] K. Oliver, W. Hou, S. Wang, Feature matching in underwater environments using sparse linear combinations, in: IEEE Conference on Computer Vision and Pattern Recognition Workshops, San Francisco, CA, USA, 2010, pp. 60–67.

[39] S. Paris, F. Durand, A topological approach to hierarchical segmentation using mean shift, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, Los Alamitos, CA, USA, 2007, pp. 1–8.

[40] R. Raveaux, J.C. Burie, J.M. Ogier, A graph matching method and a graph matching distance based on subgraph assignments, Pattern Recogn. Lett. 31 (5) (2010) 394–406.

[41] V. Robins, P.J. Wood, A.P. Sheppard, Theory and algorithms for constructing discrete Morse complexes from grayscale digital images, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2011) 1646–1658.

[42] W.R. Schwartz, R.D. da Silva, L.S. Davis, H. Pedrini, A novel feature descriptor based on the shearlet transform, in: IEEE International Conference on Image Processing, IEEE Computer Society, Brussels, Belgium, 2011, pp. 1033–1036.

[43] G.L. Scott, H.C. Longuet-Higgins, An algorithm for associating the features of two images, Roy. Soc. Lond. 244 (1309) (1991) 21–26.

[44] L.G. Shapiro, R.M. Haralick, Structural descriptions and inexact matching, IEEE Trans. Pattern Anal. Mach. Intell. 3 (5) (1981) 504–519.

[45] L.S. Shapiro, J.M. Brady, Feature-based correspondence: an eigenvector approach, Image Vision Comput. 10 (5) (1992) 283–288.

[46] R. Szeliski, Computer Vision: Algorithms and Applications, first ed., Springer-Verlag New York, Inc., New York, NY, USA, 2010.

[47] A. Torralba, K.P. Murphy, W.T. Freeman, Sharing visual features for multiclass and multiview object detection, IEEE Trans. Pattern Anal. Mach. Intell. 29 (5) (2007) 854–869.

[48] W.H. Tsai, K.S. Fu, Error-correcting isomorphisms of attributed relational graphs for pattern analysis, IEEE Trans. Syst. Man Cybern. 9 (12) (1979) 757–768.

[49] J.R. Ullmann, An algorithm for subgraph isomorphism, J. ACM 23 (1) (1976) 31–42.

[50] S. Umeyama, An eigendecomposition approach to weighted graph matching problems, IEEE Trans. Pattern Anal. Mach. Intell. 10 (5) (1988) 695–703.

[51] C. Wang, K.K. Ma, Common visual pattern discovery via directed graph model, in: IEEE ICIP, Brussels, Belgium, 2011, pp. 2957–2960.

[52] L. Wang, F. Tang, Y. Guo, S.H. Lim, N.L. Chang, Exploiting feature correspondence constraints for image recognition, in: IEEE International Conference on Image Processing, IEEE Computer Society, Brussels, Belgium, 2011, pp. 1769–1772.

[53] B. Zitová, J. Flusser, Image registration methods: a survey, Image Vision Comput. 21 (11) (2003) 977–1000.

[54] A. Zomorodian, Computational topology, in: M. Atallah, M. Blanton (Eds.), Algorithms and Theory of Computation Handbook, second ed., vol. 2, Chapman & Hall/CRC Press, Boca Raton, FL, 2010 (Chapter 3).

[55] A.J. Zomorodian, M.J. Ablowitz, S.H. Davis, E.J. Hinch, A. Iserles, J. Ockendon, P.J. Olver, Topology for Computing, Cambridge University Press, 2005.