Irregular Graph Pyramids and Representative Cocycles of Cohomology Generators[*]

Rocio Gonzalez-Diaz[1,3], Adrian Ion[3],
Mabel Iglesias-Ham[2,3], and Walter G. Kropatsch[3]

[1] Applied Math Department, University of Seville, Spain
[2] Pattern Recognition Department, CENATAV, Havana, Cuba
[3] Pattern Recognition and Image Processing Group, Vienna University of Technology

Abstract. Structural pattern recognition describes and classifies data based on the relationships of features and parts. Topological invariants, like the Euler number, characterize the structure of objects of any dimension. Cohomology can provide more refined algebraic invariants to a topological space than does homology. It assigns 'quantities' to the chains used in homology to characterize holes of any dimension. Graph pyramids can be used to describe subdivisions of the same object at multiple levels of detail. This paper presents cohomology in the context of structural pattern recognition and introduces an algorithm to efficiently compute representative cocycles (the basic elements of cohomology) in 2D using a graph pyramid. Extension to nD and application in the context of pattern recognition are discussed.

Keywords: Graph pyramids, representative cocycles of cohomology generators.

1 Introduction

Image analysis deals with digital images as input to pattern recognition systems. Topological features have the ability to ignore changes in geometry caused by different transformations. Simple features are for example the number of connected components, the number of holes, etc., while more refined ones, like homology and cohomology, characterize holes and their relations.

In order to characterize the holes in a *region adjacency graph* (RAG) associated to a 2D binary digital image, one way would be to consider the cycles with exactly 4 edges as *degenerate cycles* and establish an equivalence between all the cycles of the graph as follows: two cycles are equivalent if one can be obtained from the other by joining to it one or more degenerate cycles. There is only one equivalence class for the foreground (gray pixels) of the digital image in Fig. 1, which represents the unique hole. This is similar to consider the digital image
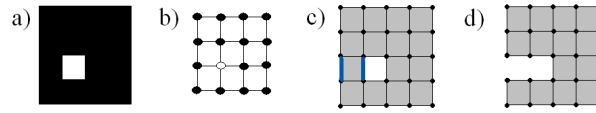
Fig. 1. a) A 2D digital image I; b) its RAG; c) a cell complex associated to I (in blue, a representative cocycle); and d) the cell complex without the hole

as a cell complex[1] [1] (see Fig. 1.c). Here one can ask for the edges we have to delete in order to 'destroy' the hole.

In the example in Fig. 1 it is not enough to delete only one edge. The set of blue edges in Fig. 1.c block any cycle that surrounds the hole; the deletion of these edges together with the faces that they bound produces the 'disappearing' of the hole. A 1-*cocycle* of a planar object can be seen as a set of edges 'blocking' the creation of cycles of one homology class. The elements of *cohomology* are equivalence classes of cocycles.

Topology simplification is an active field in geometric modeling and medical imaging (see for example [2]). In fact, the ring structure presented in cohomology is more refined than homology. The main drawbacks to using cohomology in Pattern Recognition have been its lack of geometrical meaning and the complexity for computing it. Nevertheless, concepts related to cohomology can have associated interpretations in graph theory. Having these interpretations opens the door for applying classical graph theory algorithms to compute and manipulate these features. Initial plans regarding this research have been presented in [3], and in this paper in Section 5.

The paper is organized as follows: Sections 2 and 3 recall graph pyramids and cohomology, and make initial connections. Section 4 presents the proposed method. Section 5 gives considerations regarding the usage of cohomology in image processing. Section 6 concludes the paper.

2   Irregular Graph Pyramids

A RAG, encodes the adjacency of regions in a partition. A vertex is associated to each region, vertices of neighbooring regions are connected by an edge. Classical RAGs do not contain any self-loops or parallel edges. An *extended region adjacency graph* (eRAG) is a RAG that contains the so-called *pseudo edges*, which are self-loops and parallel edges used to encode neighborhood relations to a cell completely enclosed by one or more other cells [4]. The *dual* graph of an eRAG G is called a *boundary graph* (BG) and is denoted by $\overline{G}$ (G is said to be the *primal* graph of $\overline{G}$). The edges of $\overline{G}$ represent the boundaries (borders) of the regions encoded by G, and the vertices of $\overline{G}$ represent points where boundary segments meet. G and $\overline{G}$ are planar graphs. There is a one-to-one

---

[1] Intuitively a cell complex is defined by a set of 0-cells (vertices) that bound a set of 1-cells (edges), that bound a set of 2-cells (faces), etc.
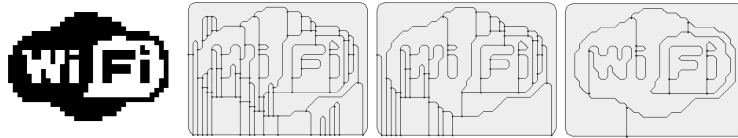
Fig. 2. A digital image I, and boundary graphs $\overline{G}_6$, $\overline{G}_{10}$ and $\overline{G}_{16}$ of the pyramid of I

correspondence between the edges of G and the edges of $\overline{G}$, which also induces a one-to-one correspondence between the vertices of G and the 2D cells (will be denoted by *faces*[2]) of $\overline{G}$. The dual of $\overline{G}$ is again G. The following operations are equivalent: edge contraction in G with edge removal in $\overline{G}$, and edge removal in G with edge contraction in $\overline{G}$.

A (dual) irregular graph pyramid [4,5] is a stack of successively reduced planar graphs $P = \{(G_0, \overline{G}_0), \ldots, (G_n, \overline{G}_n)\}$. Each level $(G_k, \overline{G}_k), 0 < k \leq n$ is obtained by first contracting edges in G_{k-1} (removal in $\overline{G}_{k-1}$), if their end vertices have the same label (regions should be merged), and then removing edges in G_{k-1} (contraction in $\overline{G}_{k-1}$) to simplify the structure. The contracted and removed edges are said to be *contracted* or *removed* (sometimes called *removal* edges) in $(G_{k-1}, \overline{G}_{k-1})$. In each G_{k-1} and $\overline{G}_{k-1}$, contracted edges form trees called *contraction kernels*. One vertex of each contraction kernel is called a *surviving vertex* and is considered to have been 'survived' to $(G_k, \overline{G}_k)$. The vertices of a contraction kernel in level $k - 1$ form the *reduction window* of the respective surviving vertex v in level k. The *receptive field* of v is the (connected) set of vertices from level 0 that have been 'merged' to v over levels $0 \ldots k$.

For each boundary graph $\overline{G}_i$, the *cell complex* [6] associated to the foreground object, called *boundary cell complex*, is obtained by taking all faces of $\overline{G}_i$ corresponding to vertices of G_i, whose receptive fields contain (only) foreground pixels, and adding all edges and vertices needed to represent the faces.

Lemma 1. *All the boundary cell complexes of a given irregular dual graph pyramid are cell subdivisions of the same object. Therefore, all these cell complexes are homeomorphic.*

As a result of Lemma 1, topological invariants computed on different levels of the pyramid are equivalent.

3   Cohomology and Integral Operators

Intuitively, homology characterizes the holes of any dimension (i.e. cavities, tunnels, etc.) of an n-dimensional object. It defines the concept of *generators* which, for example for 2D objects are similar to closed paths of edges surrounding holes. More general, k-dimensional manifolds surrounding $(k+1)$-dimensional holes are generators [6], and define equivalence classes of holes. Cohomology arises from

---

[2] Not to be confused with the vertices of the dual of a RAG (sometimes also denoted by the term *faces*).

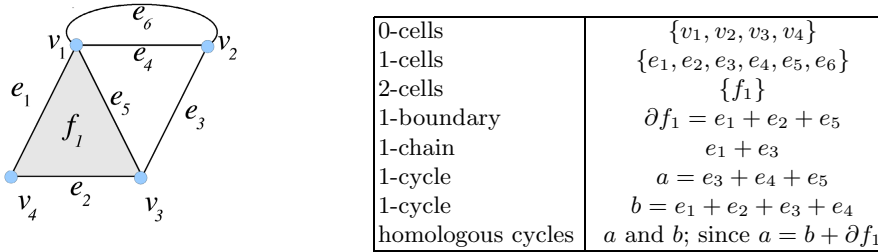| 0-cells	$\{v_1, v_2, v_3, v_4\}$
1-cells	$\{e_1, e_2, e_3, e_4, e_5, e_6\}$
2-cells	$\{f_1\}$
1-boundary	$\partial f_1 = e_1 + e_2 + e_5$
1-chain	$e_1 + e_3$
1-cycle	$a = e_3 + e_4 + e_5$
1-cycle	$b = e_1 + e_2 + e_3 + e_4$
homologous cycles	a and b; since $a = b + \partial f_1$

Fig. 3. Example cell complex

the algebraic dualization of the construction of homology. It manipulates groups of homomorphisms to define equivalence classes. Intuitively, cocycles (the invariants computed by cohomology), represent the sets of elements (e.g. edges) to be removed to destroy certain holes. See (Fig. 1.c) for an example cocycle.

Starting from a cell decomposition of an object, its homology studies incidence relations of its subdivision. Fig. 3 illustrates the following abstract concepts. A cell of dimension p is called a p-cell. The notion of p-chain is defined as a formal sum of p-cells. The chains are considered over $\mathbb{Z}/2$ coefficients i.e. a p-cell is either present in a p-chain (coefficient 1) or absent (coefficient 0) - any cell that appears twice vanishes. The set of p-chains form an abelian group called the p-chain group C_p. This group is generated by all the p-cells. The boundary operator is a set of homomorphisms $\{\partial_p : C_p \to C_{p-1}\}_{p \geq 0}$ connecting two immediate dimensions: $\cdots \overset{\partial_{p+1}}{\to} C_p \overset{\partial_p}{\to} C_{p-1} \to \cdots \overset{\partial_1}{\to} C_0 \overset{\partial_0}{\to} 0$. By linearity, the boundary of any p-chain is defined as the formal sum of the boundaries of each p-cell that appears in the chain. The boundary of 0-cells (i.e. points) is always 0. For each p, $\partial_{p-1}\partial_p = 0$. A p-chain σ is called a p-cycle if $\partial_p(\sigma) = 0$. If $\sigma = \partial_{p+1}(\mu)$ for some $(p+1)$-chain μ then σ is called a p-boundary. Two p-cycles a and a' are homologous if there exists a p-boundary b such that $a = a' + b$.

Denote the groups of p-cycles and p-boundaries by Z_p and B_p respectively. All p-boundaries are p-cycles ($B_p \subseteq Z_p$). Define the pth homology group to be the quotient group $H_p = Z_p/B_p$, for all p. Each element of H_p is a class obtained by adding each p-boundary to a given p-cycle a. Then a is a representative cycle of the homology class $a + B_p$.

Cohomology groups are constructed by turning chain groups into groups of homomorphisms and boundary operators into their dual homomorphisms. Define a p-cochain as a homomorphism $c : C_p \to \mathbb{Z}/2$. We can see a p-cochain as a binary mask of the set of p-cells: imagine you order all p-cells in the complex (let's say we have n p-cells, and call this ordered set S_p). Then a p-cochain c is a binary mask of n values in $\{0, 1\}^n$.

The p-cochains form the set C^p which is a group. A p-cochain c is totally defined by the set of p-cells that are evaluated to 1 by c. The boundary operator defines a dual homomorphism, the coboundary operator $\delta^p : C^p \to C^{p+1}$, such that $\delta^p(c) = c\partial_{p+1}$ for any p-cochain c. Since the coboundary operator runs in a direction opposite to the boundary operator, it raises the dimension. Its

kernel is the group of *cocycles* and its image is the group of *coboundaries*. Two p-cocycles c and c' are *cohomologous* if there exists a p-coboundary d such that $c = c' + d$. The pth *cohomology group* is defined as the quotient of p-cocycle modulo p-coboundary groups, $H^p = Z^p/B^p$, for all p. Each element of H^p is a class obtained by adding each p-coboundary to a given p-cocycle c. Then c is a *representative cocycle* of the cohomology class $c + B^p$. If the object is embedded in $\mathbb{R}^3$, then homology and cohomology are isomorphic. However, cohomology has a ring structure which is a more refined invariant than homology. See [6] for a more detailed explanation.

Starting from a cell decomposition of an object (e.g. from any level of the pyramid) and the chain complex associated to it, $\cdots \overset{\partial_2}{\to} C_1 \overset{\partial_1}{\to} C_0 \overset{\partial_0}{\to} 0$, take a q-cell σ and a $(q + 1)$-chain α. An *integral operator* [7] is defined as the set of homomorphisms $\{\phi_p : C_p \to C_{p+1}\}_{p \geq 0}$ such that $\phi_q(\sigma) = \alpha$, $\phi_q(\mu) = 0$ if μ is a q-cell different to σ, and for all $p \neq q$ and any p-cell γ we have $\phi_p(\gamma) = 0$. It is extended to all p-chains by linearity.

Integral operators can be seen as a kind of inverse boundary operator. They satisfy the condition $\phi_{p+1}\phi_p = 0$ for all p. An integral operator $\{\phi_p : C_p \to C_{p+1}\}_{p \geq 0}$ satisfies the *chain-homotopy property* iff $\phi_p \partial_{p+1} \phi_p = \phi_p$ for each p. For ϕ_p satisfying the chain-homotopy property, define $\pi_p = id_p + \phi_{p-1}\partial_p + \partial_{p+1}\phi_p$ where $\{id_p : C_p \to C_p\}_{p \geq 0}$ is the identity. Then, $\cdots \overset{\partial_2}{\to} im\pi_1 \overset{\partial_1}{\to} im\pi_0 \overset{\partial_0}{\to} 0$ is a chain complex and $\{\pi_p : C_p \to im\pi_p\}$ is a *chain equivalence* [6]. Its chain-homotopy inverse is the inclusion map $\{\iota_p : im\pi_p \to C_p\}$.

Consider, for example, the cell complex K in Fig. 4 on the left. The integral operator associated to the removal of the edge e is given by $\phi_1(e) = B$. Then, $\pi_1(e) = a + f + d$, $\pi_2(B) = 0$, $\pi_2(A) = A + B$ ($A + B$ is renamed as A' in K') and π_p is the identity over the other p-cells of K, $p = 0, 1, 2$. The removal of edge e decreased the degree of vertices 1 and 3 allowing for further simplification.

The following lemma guarantees the correctness of the down projection procedure given in Section 5.

Lemma 2. *Let $\{\phi_p : C_p \to C_{p+1}\}_{p \geq 0}$ be an integral operator satisfying the chain-homotopy property. The chain complexes $\cdots \overset{\partial_2}{\to} C_1 \overset{\partial_1}{\to} C_0 \overset{\partial_0}{\to} 0$ and $\cdots \overset{\partial_2}{\to} im\pi_1 \overset{\partial_1}{\to} im\pi_0 \overset{\partial_0}{\to} 0$ have isomorphic homology and cohomology groups. If $c : im\pi_p \to \mathbb{Z}/2$ is a representative p-cocycle of a cohomology generator, then $c\pi : C_p \to \mathbb{Z}/2$ is a representative p-cocycle of the same generator.*

For example, consider the cell complex K' of Fig. 4. The 1-cochain α, defined by the set $\{c, d\}$ of edges of K', is a 1-cocycle which 'represents' the white hole
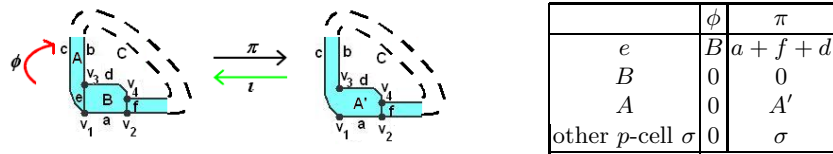

	ϕ	π
e	B	$a + f + d$
B	0	0
A	0	A'
other p-cell σ	0	σ

Fig. 4. The cell complexes K and K' and the homomorphisms ϕ, π, ι

C (in the sense that all the cycles representing the hole must contain at least one of the edges of the set). Then $\beta = \alpha\pi$ is defined by the set $\{c, d, e\}$ of edges of K. α and β are both 1-cocycles representing the same white hole C.

Lemma 3. *The two operations used to construct an irregular graph pyramid: edge removal and edge contraction, are integral operators satisfying the chain-homotopy property.*

In terms of embedded graphs an integral operator maps a vertex/point to exactly one of its incident edges and an edge to exactly one of its incident faces. In every level of a graph pyramid, the contraction kernels make up a spanning forest. A forest composed of k connected components, spanning a graph with n vertices, has k root vertices, $n - k$ other vertices, and also $n - k$ edges. These edges can be oriented toward the respective root such that each edge has a unique starting vertex. Then, integral operators mapping the starting vertices to the corresponding edge of the spanning forest can be defined as follows: $\phi_0(v_i) = e_j$, where e_j is the edge incident to v_i, oriented away from it.

Lemma 4. *All integral operators that create homeomorphisms can be represented in a dual graph pyramid. This is equivalent to: given an input image $(G_0, \overline{G}_0)$ and its associated cell complex $Z = \{C_0, C_1, C_2\}$, a cell complex $Z' = \{C'_0, C'_1, C'_2\}$ with Z, Z' homeomorphic, and Z a refinement of Z' i.e. $C'_0 \subseteq C_0$, $C'_1 \subseteq C_1$, and $C'_2 \subseteq C_2$, then there exists a pyramid P s.t. Z' is the cell complex associated to some level $(G_k, \overline{G}_k), k \geq 0$, of P.*

4   Representative Cocycles in Irregular Graph Pyramids

A method for efficiently computing representative cycles of homology generators using an irregular graph pyramid is given in [8]. In [9] a novel algorithm for correctly visualizing graph pyramids, including multiple edges and self-loops is given. This algorithm preserves the geometry and the topology of the original image.

In this paper, representative cocycles are computed and drawn in the boundary graph of any level of a given irregular graph pyramid. They are computed in the top (last) level and down projected using the described process.

For this purpose, a new level, called *homology-generator level*, is added over the boundary graph of the last level of the pyramid. The boundary graph in this new level is a set of regions surrounded by a set of self-loops incident in a single vertex. To obtain this level, on the top of the computed pyramid [8] we compute a spanning tree and contract all the edges that belong to it (see Fig. 5). Note that this last level is no longer homeomorphic to the base level, but homotopic.

Lemma 5. *The boundary cell complex of any level of the pyramid and the one of the homology-generator level have isomorphic homology and cohomology groups.*

For example, in the boundary graph of the homology-generator level (Fig. 5.a, top) each self-loop α that surrounds a region of the background (hole of a region R of the foreground) is a representative 1-cycle of a homology generator. In the same graph, the representative 1-cocycle of each cohomology generator is defined by exactly two self-loops. One of them is the self-loop α representing one homology generator. Let β be the self-loop surrounding the region R. Then, $\{\alpha, \beta\}$ is a representative 1-cocycle of a cohomology generator.

Let A_k, $k > 0$, denote the set of edges that define a cocycle in $\overline{G}_k$ (the boundary graph in level k). The *down projection* of A_k to the level $\overline{G}_{k-1}$ is the set of edges $A_{k-1} \subseteq \overline{G}_{k-1}$ that corresponds to A_k i.e. represents the same cocycle. A_{k-1} is computed as $A_{k-1} = A_{k-1}^s \cup A_{k-1}^r$, where A_{k-1}^s denotes the set of surviving edges in $\overline{G}_{k-1}$ that correspond to A_k, and A_{k-1}^r is a subset of removed edges in $\overline{G}_{k-1}$. The following steps show how to obtain A_{k-1}^r:

1. Consider the contraction kernels of G_{k-1} (RAG) whose vertices are labeled with ℓ (the region for which cocycles are computed). The edges of each contraction kernel are oriented toward the respective root - each edge has a unique starting vertex.
2. For each contraction kernel T, from the leaves of T to the root, let e be an edge of T, v its starting point, and E_v the edges in the boundary of the face associated to v: label e with the sum of the number of edges that are in both A_{k-1}^s and in E_v, and the sum of the labels of the edges of T which are incident to v.
3. A removal edge of $\overline{G}_{k-1}$ is in A_{k-1}^r if the corresponding edge of G_{k-1} is labeled with an odd number.

The proof of correctness uses the homomorphisms $\{\pi_p\}$ (proof in a follow-up journal paper).
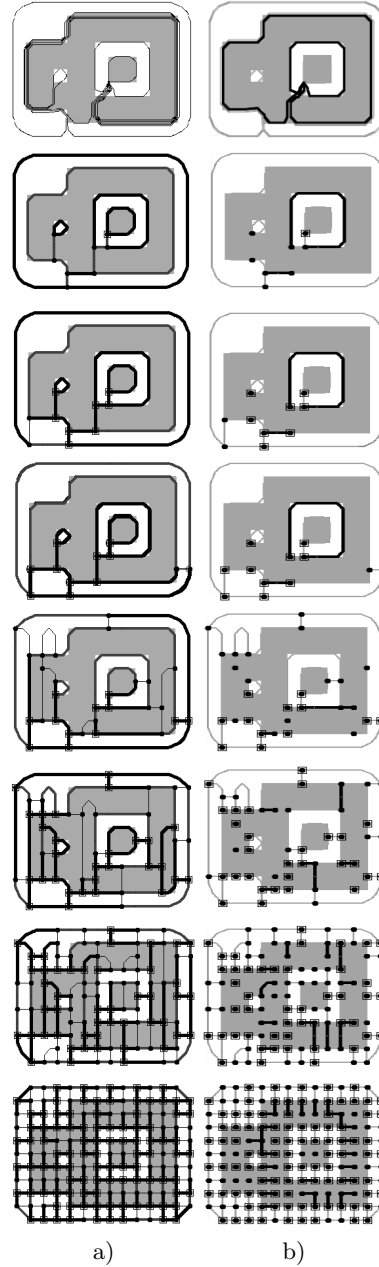

Fig. 5. a) Levels of a pyramid. Edges: removed (thin), contracted (middle) and surviving (bold). b) Down projection representative 1-cocycle (bold).

Note that these graphs were defined from the integral operators associated to the removed and contracted edges of the boundary graph of level $k-1$ to obtain level k. An example of the down projection is shown in Fig. 5.b.

Let n be the height of the pyramid (number of levels), e_n the number of edges in the top level, and v_0 the number of vertices in the base level, with $n \approx \log v_0$ (logarithmic height). An upper bound for the computation complexity is: $O(v_0 n)$, to build the pyramid; for each foreground component, $O(h)$ in the number of holes h, to choose the representative cocycles in the top level; $O(e_n n)$ to down project the cocycles (each edge is contracted or removed only once). Normally not all edges are part of cocycles, so the real complexity of down projecting a cocycle is below $O(e_n n)$. The overall computation complexity is: $O(v_0 n + c(h e_n n))$, where c is the number of cocycles that are computed and down projected.

5   Cohomology, Image Representation and Processing

Besides simplifying topology, cohomology can be considered in the context of classification and recognition based on structure. There is no concrete definition of what 'good' features are, but usually they should be stable under certain transformations, robust with respect to noise, easy to compute, and easy to match. The last two aspects motivate the following considerations: finding associations between concepts in cohomology and graph theory will open the door for applying existing efficient algorithms (e.g. shortest path); if cocycles are to be used as features for structure, the question of a stable class representative has to be considered i.e. not taking any representative cocycle, but imposing additional properties s.t. the obtained one is in most of the cases the same. The rest of the section considers one example: *1-cocycles of 2D objects*.

A 1-cocycle of a planar object can be seen as a set of edges that 'block' the creation of cycles of one homology class. Assume that the reverse is also valid i.e. all sets that 'block' the creation of cycles of one homology class are representative 1-cocycles. Then, any set of foreground edges in the boundary graph $\overline{G}_i$, associated to a path in the RAG G_i, connecting a hole of the object with the (outside) background face, is a representative 1-cocycle. It blocks any generator that would surround the hole and it can be computed efficiently (proof follows). If additional constraints are added, like minimal length, the 1-cocycle is a good candidate for pattern recognition tasks as it is invariant to the scanning of the cells, the processing order, rotation, etc.

Let K_H be the boundary cell complex associated to the foreground of the homology-generator level. Suppose that α is a representative cycle i.e. a self-loop surrounding a face of the background, and β is a self-loop surrounding a face f of the foreground such that α is in the boundary of f in K_H (Fig. 6). Let α^* denote the cocycle defined by the set $\{\alpha, \beta\}$. Let K_0 denote the boundary cell complex associated to the foreground in $\overline{G}_0$. Let ϕ be the composition of all integral operators associated with all the removals and contractions of edges of the foreground of the boundary graphs of a given irregular graph pyramid. Let $\pi = id + \phi\partial + \partial\phi$ and let $\iota : K_H \to K_0$ be the inclusion map. Consider the down projection [8] of α
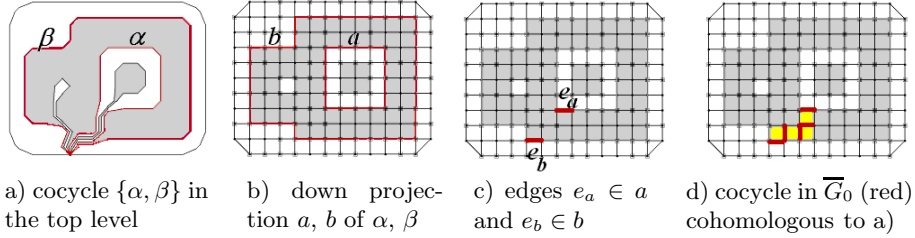
a) cocycle $\{\alpha, \beta\}$ in the top level

b) down projection a, b of α, β

c) edges $e_a \in a$ and $e_b \in b$

d) cocycle in $\overline{G}_0$ (red) cohomologous to a)

Fig. 6. Example cocycle down projection

and β in $\overline{G}_0$: the cycles $\iota(\alpha) = a$ and $\iota(\beta) = b$, respectively. Take any edge $e_a \in a$ and $e_b \in b$. Let f_a, f_b be faces of K_0 having e_a respectively e_b in their boundary. Let $v_0, v_1, \ldots, v_n$ be a simple path of vertices in G_0 s.t. all vertices are labeled as foreground. v_0 is the vertex associated to f_a, and v_n to f_b.

Proposition 1. *Consider the set of edges $c = \{e_0, \ldots, e_{n+1}\}$ of $\overline{G}_0$, where $e_0 = e_a$, $e_{n+1} = e_b$, and e_i, $i = 1 \ldots n$, is the common edge of the regions in $\overline{G}_0$ associated with the vertices v_{i-1} and v_i. c defines a cocycle cohomologous to the down projection of the cocycle α^*.*

Proof. c is a cocycle iff $c\partial$ is the null homomorphism. First, $c\partial(f_i) = c(e_i + e_{i+1}) = 1 + 1 = 0$. Second, if f is a 2-cell of K_0, $f \neq f_i$, $i = 0, \ldots, n$, then, $c\partial(f) = 0$. To prove that the cocycles c and $\alpha^*\pi$ (the down projection of α^* to the base level of the pyramid) are cohomologous, is equivalent to showing that $c\iota = \alpha^*$. We have that $c\iota(\alpha) = c(e_b) = 1$ and $c\iota(\beta) = c(e_a) = 1$. Finally, $c\iota$ over the remaining self-loops of the boundary graph of the homology-generator level is null. Therefore, $c\iota = \alpha^*$.

Observe that the cocycle c in G_0 may correspond to the path connecting two boundaries and having the minimal number of edges: 'a minimal representative cocycle'. As a descriptor for the whole object, take a set of minimal cocycles having some common property[3].

Lemma 6. *Let γ^* be a representative 1-cocycle in $\overline{G}_0$, whose projection in the homology-generator level is the cocycle α^* defined by the two self-loops $\{\alpha, \beta\}$. γ^* has to satisfy that it contains an odd number of edges of any cycle g in $\overline{G}_0$ that is homologous to $\iota(\alpha)$, the down projection of α in $\overline{G}_0$.*

Proof. γ^* contains an even number of edges of g iff $\gamma^*(g) = 1$. First, there exists a 2-chain b in K_0 such that $g = \iota(\alpha) + \partial(b)$. Second, $\gamma^*(g) = \gamma^*(\iota(\alpha) + \partial(b)) = 1$, since $\gamma^*\iota(\alpha) = \alpha^*(\alpha) = 1$, and $\gamma^*\partial(b) = 0$ because γ^* is a cocycle. So g must contain an even number of edges of the set that defines γ^*.   □

Consider the triangulation in Fig. 7, corresponding to a torus[4]. Any cycle homologous to β contains an odd number of edges of β^* (e.g. red edges in Fig. 7.c).

---

[3] E.g. they all connect the boundaries of holes with the 'outer' boundary of the object, and each of them corresponds to an edge in the inclusion tree of the object.

[4] Rectangle where bottom and top, respectively left and right edges are glued together.
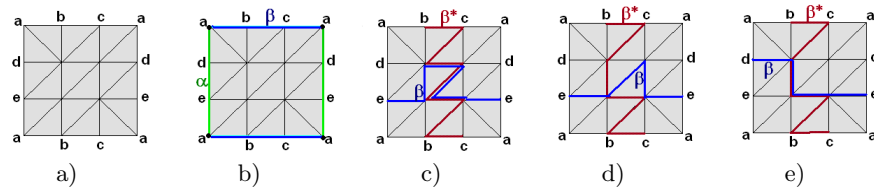
Fig. 7. A torus: a) triangulation; b) representative cycles of homology generators; c) a representative cocycle; d) and e) non-valid representative cocycles

The red edges in d) and e) do not form valid representative cocycles: in d), a cycle homologous to β (blue) does not contain any edge of β^*; in e), another cycle homologous to β contains an even number of edges of β^*.

6   Conclusion

This paper considers cohomology in the context of graph pyramids. Representative cocycles are computed at the reduced top level and down projected to the base level corresponding to the original image. Connections between cohomology and graph theory are proposed, considering the application of cohomology in the context of classification and recognition. Extension to higher dimensions, where cohomology has a richer algebraic structure than homology, and complete cohomology - graph theory associations are proposed for future work.

References

1. Hatcher, A.: Algebraic topology. Cambridge University Press, Cambridge (2002)
2. Wood, Z.J., Hoppe, H., Desbrun, M., Shroder, P.: Removing excess topology from isosurfaces. ACM Trans. Graph. 23(2), 190–208 (2004)
3. Gonzalez-Diaz, R., Iglesias-Ham, M., Ion, A., Kropatsch, W.G.: Irregular graph pyramids, integral operators and representative cocycles of cohomology generators. In: Internal Workshop on Comp. Topology in Img. Context, Poitiers, France (2008)
4. Kropatsch, W.G.: Building irregular pyramids by dual graph contraction. IEE-Proc. Vision, Image and Signal Processing 142(6), 366–374 (1995)
5. Kropatsch, W.G., Haxhimusa, Y., Pizlo, Z., Langs, G.: Vision pyramids that do not grow too high. Pattern Recognition Letters 26(3), 319–337 (2005)
6. Munkres, J.R.: Elements of Algebraic Topology. Addison-Wesley, Reading (1993)
7. González-Díaz, R., Jiménez, M.J., Medrano, B., Molina-Abril, H., Real, P.: Integral operators for computing homology generators at any dimension. In: Ruiz-Shulcloper, J., Kropatsch, W.G. (eds.) CIARP 2008. LNCS, vol. 5197, pp. 356–363. Springer, Heidelberg (2008)
8. Peltier, S., Ion, A., Kropatsch, W.G., Damiand, G., Haxhimusa, Y.: Directly computing the generators of image homology using graph pyramids. Image and Vision Computing (2008) (in press), doi:10.1016/j.imavis.2008.06.009
9. Iglesias-Ham, M., Ion, A., Kropatsch, W.G., García, E.B.: Delineating homology generators in graph pyramids. In: Ruiz-Shulcloper, J., Kropatsch, W.G. (eds.) CIARP 2008. LNCS, vol. 5197, pp. 576–584. Springer, Heidelberg (2008)