Δ-TSR: a description of spatial relationships between objects for image retrieval*

Nguyen Vu Hoang[1,2], Valérie Gouet-Brunet[2], Marta Rukoz[1,3], Maude Manouvrier[1]

1: LAMSADE - Université Paris-Dauphine - Place de Lattre de Tassigny - F75775 Paris Cedex 16

2: CEDRIC/CNAM - 292, rue Saint-Martin - F75141 Paris Cedex 03

3: Université Paris Ouest Nanterre La Défense - 200, avenue de la République - F92001 Nanterre Cedex

nguyenvu.hoang@dauphine.fr , valerie.gouet@cnam.fr , marta.rukoz@dauphine.fr, manouvrier@lamsade.dauphine.fr

August 10, 2009

Abstract

This article presents Δ-TSR, a new image content representation exploiting the spatial relationships existing between its objects of interest. This approach provides two types of descriptions: with Δ-TSR$_{3D}$, images are represented by geometric relationships between triplets of objects using triangle angles, while Δ-TSR$_{5D}$ enriches Δ-TSR$_{3D}$ by exploiting the orientation of the objects. The approach is generalized to polygonal spatial relationships, from triplets to tuples of objects. All these descriptions are invariant to translation, 2D rotation, scale and flip. A semi-local representation of the relationships is also proposed, making the description robust to viewpoint changes, if required by the application. Δ-TSR can be applied not only to symbolic images (where objects are represented by labels or icons) but also to contents represented by low-level visual features such as interest points. In this work, we evaluate Δ-TSR for image and sub-image retrieval under the query-by-example paradigm, on contents represented with interest points in a bag-of-features representation. We show that Δ-TSR improves two state-of-the-art approaches, in terms of quality of retrieval as well as of execution time. The experiments also highlight its effectiveness and scalability against large image databases.

Keywords: CBIR, spatial relationships, local image descriptors, fast retrieval

1 Introduction

Under the query-by-example paradigm, Content-Based Image Retrieval (CBIR) aims at retrieving a ranked list of similar images from an example and a set of images. One variant concerns the retrieval of sub-images or objects of interest identical or similar to a query sub-image or object. This scenario is a challenging problem due to the difficulty of representing, matching and localizing objects in potentially large-scale collections and on under viewpoint changes, partial occlusion and cluttered background. Literature on image content representation is very large, see for example [4] for a survey. In this work, we investigate approaches that focus on the representation of the spatial layout in image contents. The spatial relationships between image objects are a key point of image understanding. They offer a strong semantic which comes to enrich the low-level techniques of image visual content representation. To address this topic, many solutions have been proposed for describing

*This work is done within the French Federation WISDOM (2007-2010) and is supported by the French project ANR MDCO DISCO (2008-2010).

the spatial relationships between the objects, which can be symbolic (e.g. a tree or a person) as well as low-level features (e.g. salient points); we revisit them in section 1.1. In this work, we present Δ-TSR, which is a description of the triangular spatial relationships between objects. The main characteristics of this approach and the paper outline are summarized in section 1.2.

1.1 State of the Art

First, we summarize several propositions for describing the spatial relationships between symbolic objects in section 1.1.1. Second, because we evaluate our approach for sub-image retrieval in images where the contents are represented with sets of salient points, we give a panorama of the existing techniques representing spatial relationships between these particular objects in section 1.1.2.

1.1.1 Spatial relationships between objects in images

Among the more known categories, we can mention the directional [2, 8], topological [5], geometrical [7] and orthogonal [3] relationships. In particular, the geometrical ones lend themselves well to invariance to certain geometrical image transformations, in particular to rotation. In these approaches, the objects are represented by their centroid and metric measures between these points are exploited to characterize the relationships between couples or triplets of objects. Our work is inspired by the geometrical approach TSR (*Triangular Spatial Relationship*) which characterizes the spatial relationships between object triplets by angles built from the triangle formed by the object centroids [7, 16]. Let O_i be an object represented by the spatial coordinates of its centroid and by a code L_i of its label; two objects with different coordinates may have the same label. In the following, for the sake of simplicity, we use indistinctly O_i for the object and for its centroid. In TSR approach, the triangle formed by three objects O_1, O_2 and O_3 is represented by one of the six possible quadruplets (L_1, L_2, L_3, θ) where, L_1, L_2 and L_3 are the label codes of O_1, O_2 and O_3 respectively, and θ is the smallest angle subtended at the midpoint resulting from O_3 to the side O_1O_2 of the triangle – see Fig. 1(a). The objects are then ordered by their labels in order to always consider the same arrangement of object triplet. An image is thus encoded by the set of quadruplets representing the triangular spatial relationships between all the possible triplets of objects. This representation is invariant to rotation, translation, scaling and flipping. To minimize the storage space and speed up the search time, the continuous domain $[0°, 90°]$ associated with θ is split into D_θ classes and each quadruplet is represented by a unique and distinct single key defined by Equation 1:

$$K = D_\theta(L_1 - 1)(N_L)^2 + D_\theta(L_2 - 1)N_L + D_\theta(L_3 - 1) + (C_\theta - 1) \quad (1)$$

Where $C_\theta \in [1, D_\theta]$ is the class number to which a specific value of θ belongs, N_L the number of object labels in the database and $(L_1, L_2, L_3) \in [1, N_L]^3 \subset \mathbb{N}^3$ with $L_1 \geq L_2 \geq L_3$.

The discretization of the continuous domain of θ allows to take into account the possible variations of θ and to consider mismatch between the quadruplets representing the same objects with the same spatial relationships during matching/retrieval process. The set of keys makes thus possible to seek images and sub-images similar to a query image. Nevertheless, the number of stored keys per image is $(N_O \times (N_O - 1) \times (N_O - 2)/6) = O(N_O^3)$, with N_O the number of objects in the database. Keys are therefore stored in a B-tree. Via this structure, the authors indicate that search time is in $O(N_Q log_b N_K)$, where N_Q is the number of TSR quadruplets in the query image Q, N_K is the total number of keys in the B-tree of order b.

This approach has two drawbacks. By one hand, the tolerance interval of θ is included into the key formula of Equation 1. As a result, if the user wants to change the interval values, all keys have to be recalculated and restored. By the other hand, because of θ 's definition, even a little variation of θ can produce a large variation zone of triangles defined as similar, as shown in Fig. 1(a). This figure represents the variation zone of an equilateral triangular $O_1O_2O_3$ with a tolerance interval of θ equal to 10°. The black zone represents the

possible variations of O_1, the grey zone the possible variations of O_2 and the blue one, the variation zone of O_3: if O_2 and O_3 are fixed, all triangles formed by O_2, O_3 and all the points O_1 of the black zone are considered as similar triangles in the TSR approach.

1.1.2 Image description with Bags of Features

There exist many solutions for image description based on sets of sparse local features, see for example the recent survey on local invariant feature detectors [20]. More recently, other approaches have exploited a bag-of-features representation of the local descriptors (BoF), introduced by [19]. This concept comes from text retrieval and consists in building a visual vocabulary (a codebook) from quantized local descriptors and in representing the image by a vector of fixed size involving the frequency of each visual word of this vocabulary. The concept is usually applied to object recognition, after having trained descriptors to be tolerant to inter and intra-class variability when dealing with recognition of classes of objects. This representation has the advantage of locally describing the image content while only involving one feature vector per image. Moreover, it generates a very sparse feature space that can be accessed quickly with inverted files.

The main drawbacks of the BoF representation are that it disregards all information about the spatial layout of the local features and that the discriminative power of local descriptors is limited due both to quantization and to the large number of images. Usually, spatial information is reintroduced *a posteriori* at the last stage of retrieval on remaining matching features, by performing geometric verification or registration, as in [12] with the Hough transform or in [15] with an improved version of RANSAC. This step is particularly computational expensive, given that it is applied on a lot of features which do not verify required spatial relationships. To address this problem, the authors of [22] proposed a novel scheme to bundle the well-known SIFT features into local groups by using MSER (Maximally Stable External Region) detection. These bundled features are repeatable and much more discriminative than an individual SIFT feature. In matching bundled features, the authors define the matching score for measuring similarity of two bundled features. This score consists of membership term and a geometric term and it can be parameterized by a weighting parameter.

Incorporating spatial information *a priori* into the content description is another very relevant solution. Enriching the description with such information has the advantage of better filtering the local features to retrieve but it may remain quite challenging because of the potential expensive combinatorial description involved. Several kinds of solutions have been proposed for this category of solutions. In [18], the authors increase the visual words vocabulary with "doublets", i.e. pairs of visual words which co-occur within a local spatial neighborhood. Similar ideas were proposed in [17] with correlograms describing pairwise features in increasing neighborhoods. In [24], an invariant descriptor is proposed to model the spatial relationship of keypoints (visual words), by computing the average of the spatial distribution of a cluster center (called keyton) relative to all the keypoints of another cluster center. The relative spatial distribution is computed by separating the image's plane in co-centered circles using a keypoint as origin, each co- centered circle representing a relative distance and being broken into four equal parts. In [9] the authors propose a higher-level lexicon, i.e. visual phrase lexicon, where a visual phrase is a meaningful spatially co-occur pattern of visual words. This higher-level lexicon is less ambiguous than the lower-level one. Due to the spatial dependency of the visual data, a likelihood ratio test method is proposed to evaluate the significance of a visual word set. In addition, to overcome the complexity in searching for the meaningful patterns (the total number of possible word-sets is exponential to the cardinality of visual words lexicon), they develop an improved frequent item set mining (FIM) algorithm based on the new criteria to discover significant visual phrases in a very efficient way. In [11] the authors propose to extract progressively higher-order features, representing spatial relationships between pixels or patches, during the first-order feature (bag-of local feature descriptors) selection for object categorization. Their idea is to avoid exhaustive extraction of higher-order features and then exhaustive computation. The second-order features used are non-parametric spatial histograms describing how co-occurrences of first-order features vary in distance or with distance approximately in log scale with four directional bins (above, below, to the left and to the right).

Other approaches describe the spatial layout into a hierarchy of features. In [1], the geometry and visual appearance of objects is described as a hierarchy of parts (the lowest level being represented with local features), with probabilistic spatial relations linking parts to subparts. Similarly in [10], the pyramid match kernel of Grauman and Darrel [6] is adapted to create a spatial pyramidal image where images are recursively decomposed into sub-regions represented by BoF. The authors of [14] propose a hierarchical model that can be characterized as a constellation of BoF and that is able to combine both spatial and spatio-temporal features of videos. In [26], the authors address the tasks of detecting, segmenting and parsing deformable objects in images. A probabilistic object model is proposed, called the Hierarchical Deformable Template (HDT), which represents the object appearance at different scales over a hierarchy of structures, from elementary to more complex structures: low-level patches (gray values, gradients and Gabor filters), short-range shapes (triplets of points described by their relative angles and scales), mid-level regions (mean and variance of regions and patches) and long-range shapes characterized with triangular relationships. This set of structures were initially proposed in [25] but in [26], the learning model proposed is more sophisticated: a probability distribution is defined over the hierarchy to quantify the variability in shape and appearance of the object at multiple scales. Experimental evaluations of this model were performed mainly for detection and segmentation on public image databases (Weizmann Horse dataset and Cows). The model was also compared to state-of-the-art techniques for these tasks: among approaches that do not employ a priori information, the proposed approach provides the best results (94.7 % of segmentation accuracy).

While all the previous approaches were designed for learning of object classes for image categorization, [21] introduces, for CBIR, some spatial information into the visual description by first applying a spatial clustering on salient points in the image, providing a set of "visual constellations" that gather the points of the same neighborhood and secondly by characterizing each constellation with a classical bag-of-features representation. Note that this last approach was evaluated in terms of quality of the images retrieved, but time retrieval was not considered.

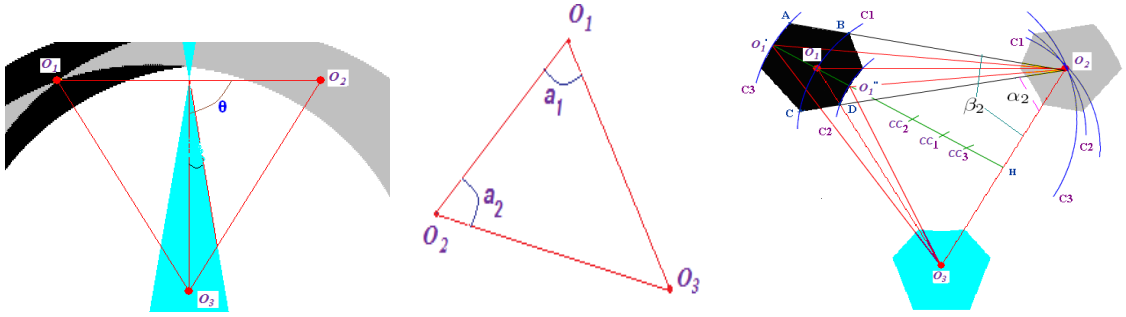
Among the different approaches presented above, the approach proposed in [26] is the only one that also exploits triplets of points as visual structures, similarly to our solution Δ-TSR. But the objectives of the authors are clearly different: in our approach, we are able to perform image or sub-image retrieval in collections of images, with a dynamic selection of the part to retrieve and without any a priori knowledge on the object or image part to be searched. Additionally, the coding of these triplets is different, with the aim of fast online retrieval of image parts in large collections, under the query-by-example paradigm.

1.2 Contributions and outline of the paper

The objective of this work is to propose an efficient and effective representation of the spatial layout of objects for image or sub-image retrieval in large collections of images, under the query-by-example paradigm. The proposed approach is called Δ-TSR and is implemented in this paper on objects which are local visual features based on salient points represented in a BoF model; it is presented in section 2. Δ-TSR uses the same idea of TSR, i.e. representing image layout by the triangular relationships between its objects. However, the adopted coding has a filtering capacity higher than TSR, leading to superior performances in terms of quality of retrieval as well as of online retrieval time. On several image data sets varying from 600 to 6000 images, we demonstrate its relevance both in terms of quality of the description (Section 3) and of retrieval time (Section 4), facing TSR and a classical bag-of-features representation of visual content. Finally, Section 5 concludes.

2 Approach Δ-TSR

In section 2.1, we present the main principles of the proposed approach for describing spatial relationships between triplets of objects in images. This approach can be enriched according to several solutions described



(a) (b) (c)

Figure 1: (a) Variation zone in TSR (b) Triangular relationship of 3 objects O_1, O_2, O_3 in Δ-TSR (c) Variation zone in Δ-TSR and its geometrical demonstration.

in section 2.2. We also generalize it to any tuples of objects in section 2.3 and then present the associated similarity measure in section 2.4. Finally, we explain our proposal for efficiently indexing this description with an access method in section 2.5.

2.1 Spatial relationship description

Taking up the idea of representing the spatial relationships between triangles of objects from the TSR approach [16], we propose a new image description called Δ-TSR$_{3D}$. This approach is applicable to objects which can be either symbolic objects represented by a point (e.g. their centroid) or low-level visual features such as interest points. Each image I of the database is represented by a set Δ-TSR$_{3D}(I)$ containing the signatures of all the triangular relationships between its objects (ordered by their labels) such as:

$$\Delta\text{-TSR}_{3D}(I) = \{S^a(O_i, O_j, O_k)/O_i, O_j, O_k \in I;\ i, j, k \in [1, N_I];\ L_i \geq L_j \geq L_k\} \tag{2}$$

with N_I the number of object in image I. The other variables are the same as those defined at Equation 1.

The triangular relationship between three objects O_i, O_j and O_k, is represented by a 3-dimensional signature:

$$S^a(O_i, O_j, O_k) = (K_1, K_2, K_3)\ ;\ \forall i, j, k \in [1, N_I] \tag{3}$$

$$\text{with} \begin{cases} K_1 = (L_i - 1)(N_L)^2 + (L_j - 1)N_L + (L_k - 1) \\ K_2 = a_i;\ K_2 \in [0°, 180°] \\ K_3 = a_j;\ K_3 \in [0°, 180°] \end{cases}$$

a_i, a_j are the angles of vertices O_i, O_j respectively – see Figure 1(b). They must satisfy the following conditions[1]:

$$\begin{cases} (a_i, a_j \in \mathbb{N}) \wedge (a_i, a_j \in [0°, 180°]) \\ L_i = L_j \Longrightarrow a_i \geq a_j \\ L_j = L_k \Longrightarrow a_i \geq 180° - a_i - a_j \end{cases}$$

K_1 coding can encompass a large number of triangles: with $N_L = 1000$ (number of labels used in most of the experiments of sections 3 and 4), K_1 can represent 1 billion of triangles, easily manageable with *integer* type. While the same key K can be associated with different triangles by setting a tolerance interval in TSR [16], in our approach each signature is associated with a single triangle and its symmetric. Moreover, instead of computing a key which depends on the tolerance interval (coded by D_θ and C_θ in the key formula of TSR – see Eq. 1), our 3-dimensional signature S^a is independent from the tolerance interval, called δ_a. Thus, changing

[1]Note that, in order to take less memory and to accelerate the process, we choose a_i and a_j in $\mathbb{N}$ instead of $\mathbb{R}$. Moreover, our experiments using real type provided no improvement.

δ_a has no impact on the image description and indexing. In return, δ_a is used to define the similarity between triangles. We consider as similar triangles of triangle T_Q all the triangles T_I whose angles verify the tolerance constraints defined by:

$$\begin{cases} \alpha_1 = max(a_i(T_Q) - \delta_a, 0^\circ) \leq a_i(T_I) \leq \beta_1 = min(a_i(T_Q) + \delta_a, 180^\circ) \\ \alpha_2 = max(a_j(T_Q) - \delta_a, 0^\circ) \leq a_j(T_I) \leq \beta_2 = min(a_j(T_Q) + \delta_a, 180^\circ) \\ a_k(T) = 180^\circ - a_j(T) - a_i(T); \quad \forall T = T_Q, T_I \\ \alpha_3 = max(a_k(T_Q) - \delta_a, 0^\circ) \leq a_k(T_I) \leq \beta_3 = min(a_k(T_Q) + \delta_a, 180^\circ) \end{cases} \tag{4}$$

These constraints can be demonstrated geometrically; we demonstrate the validity of the variation zone for O_1. Let us consider an equilateral triangle $O_1O_2O_3$ represented in Fig. 1(c). The black zone represents the possible variation zone of O_1 when O_2 and O_3 are fixed, and with $\delta_a = 10^\circ$. H represents the midpoint of O_2O_3, $O_{1'}$ corresponds to the point such as $\widehat{O_2O_{1'}O_3} = \alpha_1$ and $O_{1''}$ to the point such as $\widehat{O_2O_{1''}O_3} = \beta_1$. Let C_1, C_2, C_3 be the arcs of circles circumscribing triangles $O_1O_2O_3$, $O_{1'}O_2O_3$ and $O_{1"}O_2O_3$. Every point P on arcs $C1$ is such that $\widehat{O_2MO_3} = \widehat{O_2O_1O_3}$. Therefore, all arcs of circles with a center CC on the median O_1H and which pass through the area bounded by arcs of circles C_2 and C_3 contain M such that $\alpha_1 \leq \widehat{O_2MO_3} \beta_1$. This represents the set E_1 of points O_1 that validates the first constraint of tolerance above – see the first line in Eq. (4). All triangles similar to $O_1O_2O_3$ must also satisfy the second constraint of tolerance. Let O_2B and O_2D be the lines such that $\widehat{BO_2O_3} = \beta_2$ et $\widehat{DO_2O_3} = \alpha_2$. Then, all points O_1 in the area bounded by these 2 lines verify $\alpha_2 \leq \widehat{O_1O_2O_3} \leq \beta_2$. This corresponds to the set E_2 of points validating the second constraint. In the same way, we can find set E_3 of points validating the third condition of tolerance. As a result, the intersection of E_1, E_2 and E_3 represents a (black) region containing all points O_1 which allow to construct a triangle similar to the initial triangle $O_1O_2O_3$. As $O_1O_2O_3$ is equilateral, areas of variation of the 3 vertices are identical. For another type of triangle, we can also define its areas of geometrical variations.

Note that if angles are not considered in the signature, we obtain a signature Δ-TSR$_{1D}$, similar to those proposed in [17, 18] (see section 1.1.2), but which considers co-occurrences of triplets instead of doublets of objects, such as:

$$\Delta\text{-TSR}_{1D}(I) = \{S^\ell(O_i, O_j, O_k) / O_i, O_j, O_k \in I;\ i, j, k \in [1, N_I];\ L_i \geq L_j \geq L_k\} \tag{5}$$

$$\text{with:} \quad S^\ell(O_i, O_j, O_k) = (K_1)\ ;\ \forall i, j, k \in [1, N_I] \tag{6}$$

2.2 Improvements of the description

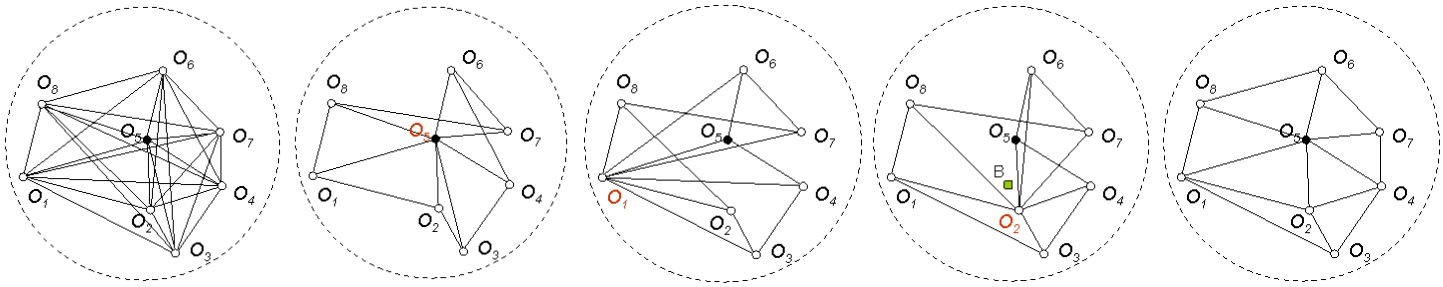
Δ-TSR$_{3D}$ can be improved or adapted according to the application. We present two possible improvements in sections 2.2.1 and 2.2.2.

2.2.1 Description Δ-TSR$_{5D}$

In Δ-TSR$_{3D}$, the objects of interest are exclusively represented by their centroids. For a large palette of objects, it is possible to consider an *orientation*: if the object is symbolic, we can consider the orientation of its largest segment or the average orientation computed on points inside the object; with salient points, a main orientation can be computed from the local gradient around the point, as in the SIFT description for example [12].

By representing an object by its centroid plus its orientation, we propose the representation Δ-TSR$_{5D}$, which is also invariant to translation, 2D rotation, scale and flip. The triangular relationship of 3 objects O_i, O_j, O_k whose digital labels are L_i, L_j, L_k can be represented by a 5-tuple $(K_1, K_2, K_3, K_4, K_5)$, where triplet (K_1, K_2, K_3) corresponds to $S^a(O_i, O_j, O_k)$ of Equation 3. Let γ_l be the orientation's angle associated with O_l (with respect to x axis for simplicity). K_4 and K_5 represent the relative orientation of O_i and O_j with respect to O_k, in order to maintain invariance to 2D rotation. The associated description, S^o, is described by:

$$S^o(O_i, O_j, O_k) = (K_1, K_2, K_3, K_4, K_5) \tag{7}$$



SL (56 triangles) $\text{SL}_{sem+cur}$ (7 triangles) $\text{SL}_{sem+1st}$ (7 triangles) $\text{SL}_{sem+bar}$ (7 triangles) SL_{geo} (8 triangles)

Figure 2: Illustration of the pruning strategies on a semi-local neighborhood centered on object O_5 and containing 8 objects. The value in parentheses is the number of triangles involved. For SL_{sem}, the red point is the pivot (O_5 for $\text{SL}_{sem+cur}$, O_1 for $\text{SL}_{sem+1st}$ and O_2 for $\text{SL}_{sem+bar}$). In the $\text{SL}_{sem+bar}$ triangulation, the squared green point "B" is the barycenter of the 8 centroids.

$$\text{with} \begin{cases} K_1 = (L_i - 1)(N_L)^2 + (L_j - 1)N_L + (L_k - 1) \\ K_2 = a_i;\ K_2 \in [0°, 180°] \\ K_3 = a_j;\ K_3 \in [0°, 180°] \\ K_4 = |\gamma_i - \gamma_k|;\ \gamma_i, \gamma_j, \gamma_k \in [-180°, 180°] \\ K_5 = |\gamma_j - \gamma_k|;\ K_4, K_5 \in [0°, 360°] \end{cases}$$

After δ_a, a second threshold δ_o is introduced for considering similarity search with K_4 and K_5, $K_4(T_I)$ varying in $[K_4(T_Q) - \delta_o, K_4(T_Q) + \delta_o]$ and $K_5(T_I)$ in $[K_5(T_Q) - \delta_o, K_5(T_Q) + \delta_o]$.

Finally, the image's global signature, noted $\Delta\text{-TSR}_{5D}(I)$, is obtained similarly to $\Delta\text{-TSR}_{3D}(I)$:

$$\Delta\text{-TSR}_{5D}(I) = \{S^o(O_i, O_j, O_k) / O_i, O_j, O_k \in I;\ i, j, k \in [1, N_I];\ L_i \geq L_j \geq L_k\}$$

2.2.2 Selection of relevant triplets of objects

$\Delta\text{-TSR}_{1D}$, $\Delta\text{-TSR}_{3D}$ and $\Delta\text{-TSR}_{5D}$ may involve triplets of objects located far away from each other in the image. Such a representation is adequate for global image content description and retrieval, but is not for sub-image or object of interest retrieval. Here, a *semi-local description* of the spatial layout, that privileges smaller triangles of objects, is largely sufficient for sub-image querying and ensures more robustness to changes of viewpoint for retrieval of objects of interest represented with several triangles; objects described with BoF typically enter in this category. Let r be a radius of neighborhood; for sake of simplicity, r is a predefined parameter fixed for all objects, but it can be adapted given an idea of object's scale, such as with the SIFT points which are extracted at specific scales. The following strategy of triangle selection is applied for each object O_i of image I:

Strategy SL: Semi-local pruning of triplets

1. Find all objects O_j in the neighborhood of O_i, i.e. $d_{L_2}(O_i, O_j) \leq r$ where d_{L_2} stands for the Euclidean distance;
2. Build all the triangular relationships from the list of items $\{O_j\}$ found.

One interesting consequence of this strategy is that the number of triangles of the image signature ($C^3_{N_I}$ by default, where N_I is the number of objects in I) is considerably reduced to an average amount of $N_I \times C^3_{\bar{n}}$ triplets, where $\bar{n}$ is the average number of objects in each neighborhood. With SL, all the possible triangular

relationships are build in a semi-local neighborhood. In each neighborhood, we can reduce the complexity of the description even more by adding other strategies of pruning, such as the two following ones:

Strategy SL$_{sem}$: Pruning based on semantics

This strategy is also semi-local and keeps a triangulation that is deduced from the object's labels, as follows:

1. Find all objects O_j in the neighborhood of O_i, as in strategy SL;
2. Sort list $\{O_j\}$ by increasing order of their labels L_i ;
3. Select in list $\{O_j\}$ a particular object O_p (pivot), according to a criteria of selection defined below. Remove O_p from $\{O_j\}$;
4. From this list of size $|\{O_j\}|$, build all the triangular relationships (O_p,O_{j_l},O_{j_m}) where $l = 1, .., |\{O_j\}|$, $m = 1$ if $l = |\{O_j\}|$ and $m = l + 1$ otherwise.

We propose three alternatives for selecting pivot O_p:

- Strategy SL$_{sem+cur}$: Consider O_i as O_p;
- Strategy SL$_{sem+1st}$: Take O_p as the first object of list $\{O_j\}$ (remember that the list is sorted on the labels);
- Strategy SL$_{sem+bar}$: Take O_p as the object closest to the spatial barycenter of items in $\{O_j\}$. This strategy is not purely based on semantics since it exploits the geometry of the objects by the way of their barycenter.

Whatever the criteria of selection of O_p, strategy SL$_{sem}$ ensures that the size of the image signature is reduced to an average of $N_I \times (\bar{n} - 1)$ triplets. Inside each neighborhood, we obtain a minimal set of triangles, potentially overlapping, that connects every object O_j to three other objects at least.

Strategy SL$_{geo}$: Pruning based on geometry

Differently to strategy SL$_{sem}$, this strategy provides a triangulation of the objects in the semi-local neighborhood, that is directly deduced from their geometry, as follows:

1. Find all objects O_j in the neighborhood of O_i, as in strategy SL;
2. From all the possible triangles built on set $\{O_j\}$, select the subset that form a Delaunay triangulation.

A Delaunay triangulation is chosen for this strategy, because it maximizes the minimum angle of the involved triangles, thus reducing their stretching and then increasing their robustness to viewpoint changes by preserving locality. Such a triangulation ensures that the size of the image signature is reduced to an average of $N_I \times [2(\bar{n} - 1) - \bar{e}]$ triplets, where $\bar{e}$ is the average number of objects on the convex envelope. Inside each neighborhood, we obtain a minimal set of disjoint triangles, that form a partition of the convex envelope associated with objects $\{O_j\}$.

Strategies SL, SL$_{sem}$ and SL$_{geo}$ are illustrated in Figure 2 on a set of 8 objects. They are evaluated and discussed in section 4.2.

2.3 Generalization of Δ-TSR: Δ-PSR

"PSR" is the abbreviation for "Polygonal Spatial Relationships", which generalizes Δ-TSR to polygonal supports of description of the spatial layout. If image I contains N_I objects, we consider d objects $O_1, ..., O_d$ of I $(3 \leq d \leq N_I)$, which are organized to verify $L_1 \geq ... \geq L_{d-1} \geq L_d$. Then, the relationship between these objects is represented by a closed polygon composed of vertices $(O_1, ..., O_d)$ where consecutive vertices are connected. Let a_i be the angle associated with each of these vertices, and γ_i the orientation of O_i. Note that if $L_i = L_j$ for two objects O_i and O_j, they are sorted according to these angles and orientations. According to the information considered, the corresponding signatures of polygons are defined as follows:

$$\begin{cases} S^{\ell}(O_1, ..., O_d) = (K_1) \\ S^{a}(O_1, ..., O_d) = (K_1, K_2, ..., K_d) \\ S^{o}(O_1, ..., O_d) = (K_1, K_2, ..., K_d, K_{d+1}, ..., K_{2d-1}) \end{cases} \tag{8}$$

$$\text{with} \begin{cases} K_1 = \sum_{i=1}^{d} (L_i - 1) \times (N_L)^{d-i} \\ K_2 = a_1; \;\; K_2 \in [0°, 180°] \\ K_3 = a_2; \;\; K_3 \in [0°, 180°] \\ ... \\ K_d = a_{d-1}; \;\; K_d \in [0°, 180°] \\ K_{d+1} = |\gamma_1 - \gamma_d|; \;\; K_{d+1} \in [0°, 360°] \\ K_{d+2} = |\gamma_2 - \gamma_d|; \;\; K_{d+2} \in [0°, 360°] \\ ... \\ K_{2d-1} = |\gamma_{d-1} - \gamma_d|; \;\; K_{2d-1} \in [0°, 360°] \end{cases} \tag{9}$$

Note that the higher d and N_L, the higher the value of K_1. Classical types of data being constrained, this coding may increase the complexity of the implementation for representing this key numerically.

Finally, the global signature Δ-PSR associated with image I follows model Δ-PSR(I) $= \{S^*(\mathcal{A}_d)\}$, knowing that $\mathcal{A}_d$ is any subset of d objects of I. It is Δ-PSR$_{1D}$(I) with S^{ℓ}, Δ-PSR$_{dD}$(I) with S^a and Δ-PSR$_{(2d-1)D}$(I) with S^o. The number of subsets might be reduced if pruning rules SL and SL$_{sem}$ defined in section 2.2.2 are considered.

2.4 Similarity measure

With Δ-PSR, the similarity between two images can be established by the ratio of similar signatures between them. Let P_Q be a polygon composed of vertices O_1,...,O_d of image query Q and P_I a polygon composed of vertices O'_1,...,O'_d of an image I, such as objects O_i and O'_i have the same label L_i $(i \in [1, N_I])$. Each image of the database is represented by a collection of signatures $S^*(P_I)$ thus the image retrieval problem becomes the problem of matching between the polygon signatures $S^*(P_Q)$ and $S^*(P_I)$ such that $K_1(P_Q) = K_1(P_I)$ taking into account the tolerance intervals δ_a and δ_o. To enhance this purpose, we propose a similarity measure between images, SIM, based on a similarity measure between polygon signatures, sim. These measures vary in the interval $[0, 1]$ and increase with the similarity.

Signature similarity measure (sim): the similarity between $S^*(P_Q)$ and $S^*(P_I)$ is defined by Equation 10, with $(S^*(P_Q), S^*(P_I)) \in \{(S^a(P_Q), S^a(P_I)), (S^o(P_Q), S^o(P_I))\}$.

$$sim(S^*(P_Q), S^*(P_I)) = \begin{cases} sim^a(S^*(P_Q), S^*(P_I)) + \\ sim^o(S^*(P_Q), S^*(P_I)) \\ \text{if } S^*(P_I) \text{validates the tolerance intervals} \\ \\ 0 \text{ otherwise} \end{cases} \tag{10}$$

with sim^a and sim^o defined as:

$$sim^a(S^*(P_Q), S^*(P_I)) = \begin{cases} 1 \text{ if } \delta_a = 0 \\ \\ \frac{1}{d-1}(\sum_{i=2}^{d}(1 - \frac{|K_i(P_Q) - K_i(P_I)|}{\delta_a})) \\ \text{if } \delta_a \neq 0 \end{cases}$$

$$sim^o(S^*(P_Q), S^*(P_I)) = \begin{cases} 0 \text{ if orientation is not taken into account} \\ \\ 1 \text{ if } \delta_o = 0 \\ \\ \frac{1}{d-1}(\sum_{i=d+1}^{2\times d-1}(1 - \frac{|K_i(P_Q) - K_i(P_I)|}{\delta_o})) \\ \text{if } \delta_o \neq 0 \end{cases}$$

Image similarity measure (SIM): let Δ-PSR(I) and Δ-PSR(Q) be the global signatures associated with images Q and I respectively, and SP(Q,I) the set of couples $(S^*(P_Q), S^*(P_I))$ involving most similar polygons P_Q of Q and P_I of I, such as:

$$SP(Q,I) = \left\{ \begin{array}{l} (S^*(P_Q), S^*(P_I)) \in \{(S^a(P_Q), S^a(P_I)), (S^o(P_Q), S^o(P_I))\} \ / \\ S^*(P_Q) \in \Delta\text{-PSR}(Q) \ \wedge S^*(P_I) \in \Delta\text{-PSR}(I) \ \wedge sim(S^*(P_Q), S^*(P_I)) \neq 0 \wedge \\ sim(S^*(P_Q), S^*(P_I)) = \max\limits_{\forall S^*(P'_I) \in \Delta\text{-PSR}(I)} (sim(S^*(P_Q), S^*(P'_I)) \wedge \\ sim(S^*(P_Q), S^*(P_I)) = \max\limits_{\forall S^*(P'_Q) \in \Delta\text{-PSR}(Q)} (sim(S^*(P'_Q), S^*(P_I)) \end{array} \right\} \quad (11)$$

The similarity between images Q and I is then defined as follows:

$$SIM(Q,I) = \frac{\sum_{k=1}^{card(SP(Q,I))} sim(SP_k(Q,I))}{card(\Delta\text{-PSR}(Q))} \quad (12)$$

where $SP_k(Q,I)$ is the k^{th} item of $SP(Q,I)$. The resulting images are ordered based on the SIM similarity measure.

2.5 Associated access method

As explained in section 2.4, the similarity image retrieval process requires comparing all the descriptors signatures of the query image with the descriptors signatures of each image stored in the database to calculate their similarity measure. Like in [16], we propose to use a structure indexing all polygon signatures. Each polygon signature is associated with its corresponding image. To find the polygon signatures similar to a signature $S^*(P_Q) = (K_1(P_Q), .., K_{2d-1}(P_Q))$ the search process is the following one:

1. Search for all the signatures having a key K_1 equal to $K_1(P_Q)$;
2. Select signatures $S^*(P_I)$, found in step 1, that validate the tolerance intervals on angles (δ_a);
3. Select signatures of step 2 that validate the orientation tolerance intervals (δ_o);
4. Compute $sim(S^*(P_Q), S^*(P_I))$.

If the polygon signatures are ordered, in such a way that $S^*(P_I) > S^*(P_Q)$ if and only if $\exists i \ / \ K_i(P_I) > K_i(P_Q) \ \wedge \ \forall j < i \ K_j(P_I) = K_j(P_Q)$, then the searching process becomes to search the set of signatures S_I^* in the interval $[BI_i, BI_f]$ where $BI_i = (K_1, K_2 - \delta_a, .., K_d - \delta_a, K_{d+1} - \delta_o, .., K_{2d-1} - \delta_o)$ and $BI_f = (K_1, K_2 + \delta_a, .., K_d + \delta_a, K_{d+1} + \delta_o, .., K_{2d-1} + \delta_o)$. Consequently it is optimal to use the classical searching

structure B-tree to index the composite searching key $S^*(P)$. In this way the searching time is $O(N_{AP} log_b N_P)$ where N_{AP} is the average number of polygon in the images, N_P is the number of polygon in the database and b is the order of the B-tree. Since the searching key $S^*(P)$ is multidimensional, we have also experimented indexing using the classical multidimensional structure R-tree. However the results do not present any improvement regarding the B-tree ones, as shown in Section 4.1.

3 Qualitative evaluation of Δ-TSR

This section and section 4 are dedicated to the evaluation of Δ-TSR for sub-image / object of interest retrieval by example in a collection of images. The sets of objects O_i representing the image contents are visual words that correspond to interest points in a BoF representation. We compare the performances of Δ-TSR to TSR [16] and to the classical BoF representation of an image [19].

3.1 Framework of the evaluation

Material

All the approaches were developed in Java. The tests were performed on a PC with an Intel Core 2 2.17GHz and 4GB RAM, running under Windows XP.

Databases

For the main experiments, we consider two databases of images: one of 6000 images, noted DB_{6000}, and the second with 600 images, noted DB_{600}, which is a subset of DB_{6000}. Each image contains an object from the well-known data set *COIL-100*[2], synthetically inserted on a photograph as background (images of 352×288 pixels with heterogeneous content downloaded from Internet). In DB_{6000}, we consider 6000 different backgrounds and 100 objects under 6 different 3D poses, inserted with a specific 2D rotation per pose on 10 backgrounds; there are 60 images per object and 10 images per 2D rotation/3D pose, as illustrated in Figure 3. DB_{600} is obtained by considering 20 random objects under 3 different 3D poses / 2D rotations and 10 backgrounds, leading to 30 images per object.

Figure 3: Samples of DB_{6000}: two objects with different backgrounds and 3D poses / 2D rotations.

Visual words as labeled objects

As objects O_i used in all the evaluated techniques, we have chosen interest points in a bag-of-features representation, that attaches a category (a visual word) to each point. Points were extracted with the Harris color detector

[2]http://www1.cs.columbia.edu/CAVE/software/softlib/coil-100.php.

and locally described with color differential invariants [13]; other descriptors, SIFT for example [12], could also be considered without challenging the relevance of the approach. In DB_{6000}, the number of extracted points varies between 300 and 600 points per image. The visual vocabulary was classically obtained using k-means as clustering algorithm [19], and its size (N_L) was varied from 250 to 2000 words during the experiences.

Implementation of compared techniques

The TSR technique implemented follows work [7, 16] presented in section 1.1.1. It is applied on visual words and each key K is indexed in a B-tree. To be able to compare this approach with the semi-local version of Δ-TSR (see section 2.2.2), we have also adapted TSR to deal with a subset of all the possible triangles, and then adapted the similarity measure: this version is called TSR_{sl}. The bag-of-features representation implemented follows the classical one initially proposed in [19]: as associated image signature, we employ the standard weighting known as "term frequency-inverse document frequency" (*tf-idf*) to compute the histogram of the visual vocabulary. Each signature is indexed according to an inverted file to accelerate retrieval. Image signatures are compared with their normalized scalar product (cosine of angle). Note that a great effort was made to implement these state-of-the-art techniques optimally.

Criteria of evaluation

All the techniques are evaluated both in terms (1) of quality of the responses by computing Precision and Recall (P/R) curves (results presented in this section by displaying these curves or at least mAP, i.e. mean Average Precision for secondary results) and (2) of time retrieval by measuring CPU time and IO access (see Section 4). These measures are averaged over all the images of the tested database (DB_{600} or DB_{6000}) taken as queries. Note that here, to precisely evaluate our contribution on the description of the spatial layout of visual contents, we did not end retrieval by performing a *a posteriori* geometric registration of the matched features to re-rank the responses, whatever the evaluated approach.

3.2 Comparison of Δ-TSR with literature

In this section, we compare Δ-TSR_{3D} (section 2.1) to Δ-TSR_{5D} (section 2.2.1), to TSR and to the classical BoF representation of an image. We also evaluate its variant Δ-TSR_{1D} that only describes co-occurrences of objects without exploiting their geometry.

All the descriptions are built on a visual vocabulary of size $N_L = 1000$. Because the evaluation is performed for sub-image / object retrieval on databases of images containing 3D objects (see section 3.1), we evaluate TSR, Δ-TSR_{1D}, Δ-TSR_{3D}, Δ-TSR_{5D} in their semi-local representation (strategy SL presented in section 2.2.2), noted TSR_{sl} for TSR. Preliminary experiments have to be done to fix the parameters associated with these approaches: radius r of the neighborhood considered in the semi-local description of each approach, the tolerance thresholds δ_a for Δ-TSR_{3D}, δ_a and δ_o for Δ-TSR_{5D}, D_θ for TSR_{sl}. Table 1 reports the average precisions obtained for optimal values of δ_a and D_θ with several values of r.

Whatever the approach, the best results are obtained with $r = 8$, and with $\delta_a = 14°$ for Δ-TSR_{3D} and $D_\theta = 1$ for TSR_{sl}. Retrieval becomes worse when the radius increases, thus demonstrating the relevance of the semi-local representation for sub-image / object retrieval. Considering larger triangles into the description leads to an image representation less efficient because (i) less robust to viewpoint changes and (ii) involving more triplets of points overlapping object of interest and background, thus less repeatable across images. One interesting consequence of this result is that, with a small radius, the complexity of Δ-TSR_{3D} as well as TSR_{sl} is notably reduced: for $r = 8$ on DB_{600}, we obtain 301,091 triangles, while it is 11,300,207 when $r = 32$. With TSR_{sl}, the optimal value observed for D_θ is 1, meaning that only one class of angles is applied on angle θ. This result indicates that the best performance is obtained without applying any constraint on this angle; it confirms

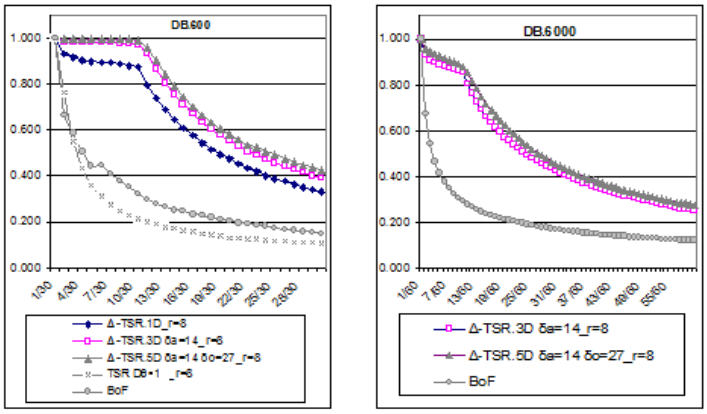
	Δ-TSR$_{3D}$		TSR$_{sl}$	
r	mAP	δ_a	mAP	D_θ
8	0.721	14°	0.242	1
16	0.716	11°	0.229	1
24	0.683	13°	0.148	1
32	0.679	10°	0.121	1

Table 1: Evaluation of Δ-TSR$_{3D}$ and TSR$_{sl}$ on DB$_{600}$ by varying radius r: mean Average Precision (mAP) for Δ-TSR$_{3D}$ with optimal δ_a for each r and for TSR with optimal D_θ for each r.

that the associated description, already illustrated in Figure 1(a), is not adapted for object retrieval with local visual features.

Δ-TSR$_{5D}$ is used by considering the color gaussian gradient around the point as orientation. With the best parameters found for Δ-TSR$_{3D}$ ($\delta_a = 14°$, $r = 8$) and by varying the orientation variation interval δ_o, on DB$_{600}$ with $N_L = 1000$, the best results for Δ-TSR$_{5D}$ are obtained with $\delta_o = 27°$.

According to these results, Figure 4(a) plots the P/R curves obtained for all the approaches, with the best values of parameters r, δ_a, δ_o and D_θ, and with strategy of triangle pruning SL. These measures were computed on DB$_{600}$, comparable results were obtained on DB$_{6000}$.



(a) (b)

Figure 4: Comparison of the best configuration of approaches (a) on DB$_{600}$ and (b) on DB$_{6000}$.

First, we observe that employing Δ-TSR$_{1D}$ greatly improves retrieval with BoF only: as already studied in works [17, 18] with doublets of visual words, characterizing the semi-local co-occurrence of objects (here with triplets of objects) produces a distinctive description of the visual content. By characterizing the geometry of triplets of objects, Δ-TSR$_{3D}$ brings a relevant information that increases precision even more whatever the recall. The Δ-TSR$_{5D}$ description slightly enriches the Δ-TSR$_{3D}$ one by exploiting the object orientation. As shown on the P/R curves of Figure 4(b), these improvements are also repeated with database DB$_{6000}$, thus confirming the usefulness of integrating the object orientation into the image signature.

Second, results obtained with TSR$_{sl}$ confirm that this approach is not adequate for sub-image retrieval. In particular, choosing $D_\theta = 1$ comes down to only considering co-occurrences of triplets, as with Δ-TSR$_{1D}$. The difference in the performance with Δ-TSR$_{1D}$ is due to the similarity measure associated with TSR. We have experimented that applying our similarity measure (see Equation 12) provides a P/R curve similar to the Δ-TSR$_{1D}$ one.

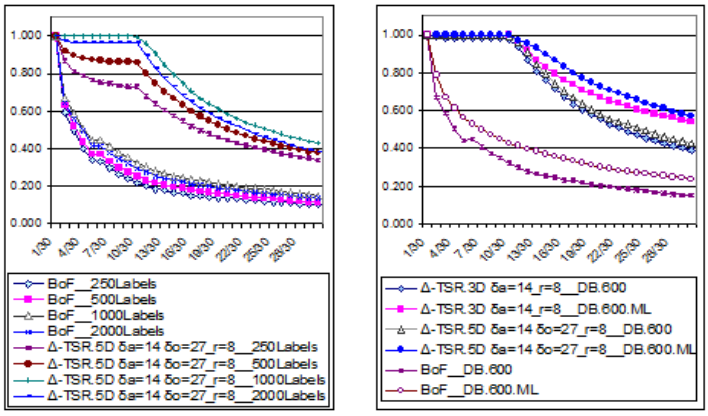
Finally, note that the shapes of the three best curves related to Δ-TSR highlight the relative sensibility of the

descriptions to viewpoint changes: from recall 11/30 where viewpoint is modified, precision decreases. Indeed, viewpoint changes may lead to the disappearance/appearance of interest points that imply the modification of the set of point triplets across images.

3.3 Influence of the labels

All the experiments presented above were done with $N_L = 1000$ and with a vocabulary built with k-means approach as explained in Section 3.1. Here, we show how the size and the quality labels categorization can influence the quality of the Δ-TSR responses. All the experiments were done with the best configurations of all the approaches on DB_{600}.

Influence of the vocabulary size. The vocabulary size has an effect on the discriminating power and the generalization of the signatures [23]. With a small size, the visual word features are not discriminative: dissimilar objects may be associated with the same word. As the size increases, they becomes more discriminative. In return, it becomes less generalizable and could be prone to noise such that two similar objects can be represented by different words. As shown in Figure 5(a), the influence of N_L is the same for all approaches (BoF and Δ-TSR_{5D}). From 250 to 1000 labels, the result quality increases with N_L. However, after $N_L = 1000$, the quality decreases. Our experiments, also confirmed on DB_{6000}, show that the best vocabulary size is 1000 labels for the two studied approaches.



(a) (b)

Figure 5: Influence (a) of vocabulary size and (b) of vocabulary quality.

Influence of the vocabulary quality. The change on 3D pose can decrease the visual word categorization quality. Indeed, analyzing the object labels, we found that changes on 3D involve noise and then the same salient point in two images of the same object with different 3D poses can have a different label in each image. In order to improve the label quality for these experiments, we have corrected the labels by tracking the object on its trajectory through the different 3D poses. Figure 5(b) compares each approach using the original labels and the modified (corrected) labels (called ML in the figure). The result quality is improved for each approach when the labels are corrected. Moreover, the gap between Δ-TSR_{3D} and Δ-TSR_{5D} increases with the corrected labels. Indeed, the better the quality of labels, the better the impact of the orientation in the image signature.

3.4 Evaluation of Δ-PSR

As explained in Section 2.3, our approach can represent the geometrical relationships between polygons of objects. In order to evaluate the best polygon degree, we have done several experiments on database DB_{600} with the corrected labels. Note that the size of the polygon signature increases with the polygon degree: for example, each polygon signature is a 7-tuple when the polygon degree is 4. As shown on Figure 6, the best polygon degree is 3. The figure only shows the results obtained with Δ-$4SR_{7D}$, when the polygon degree is 4. With a higher degree, the results are worst. Indeed, the bigger the polygon degree, the bigger the signature size and the lower the probability to find similar polygons. That is why in the following experiments, we only use triangular relationships between objects.

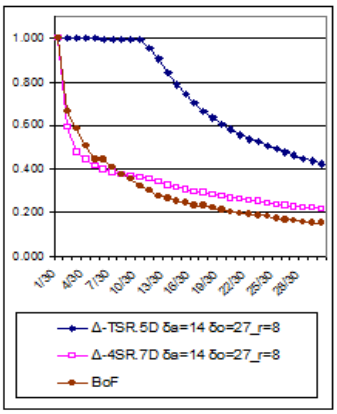


Figure 6: Evaluation of Δ-PSR: comparison between Δ-TSR_{5D}, Δ-$4SR_{7D}$ and BoF on DB_{600}.

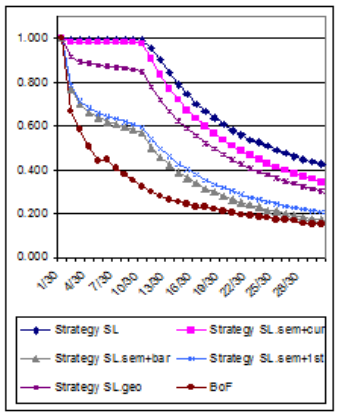


Figure 7: P/R curves obtained with the different strategies of pruning with Δ-TSR_{5D} on DB_{600}.

4 Scalability of Δ-TSR

In this section, we evaluate the performances of Δ-TSR in terms of time retrieval by measuring CPU and IO times. The experiences are done with the best configurations of the methods with the real labels, determined in section 3. All the measures are averaged over all the images of the tested database (DB_{600} or DB_{6000}) taken as queries. To obtain a realistic estimation of time retrieval, measures include the computation of the description on each query image (except the step of objects extraction, i.e. interest points detection, description and categorization, which is common to all the approaches): for the image signature based on BoF, these measures include the computation of the histogram of the visual vocabulary, while they include all the relationships computation and description for TSR and Δ-TSR.

4.1 Time retrieval estimation

In order to improve time retrieval for each approach, we have developed several index structures. The best index structure for BoF representation [19] is the inverted file. In the TSR approach, because the triangle signatures are mono-dimensional, the authors use the well-known B-tree. In our approach, the triangle signatures are multidimensional and because they can be ordered according to the searching algorithm, as explained in Section 2.5, we also experimented the B-tree index with the triangle signature as the composite searching key. Nevertheless, because the signature is multidimensional, we also experimented a classical multidimensional indexing structure, the R-tree[3], reputed to be efficient for range queries.

[3]Index taken from `http://www.rtreeportal.org`.

Table 2 shows the CPU and IO times of the different approaches. Using the small database DB_{600}, the BoF description with the inverted file is the quickest approach. Experimented on the same database, the time obtained with TSR_{sl} and Δ-TSR is equivalent when our approach is indexed by a B-tree. Using a R-tree, we obtained the worst times. Indeed, as explained in Section 2.4, the search process of Δ-TSR is based on a key ranking. For such kind of particular search query, the R-tree is not the optimal index structure.

	DB_{600}		DB_{6000}	
Approach / Index	CPU	IO	CPU	IO
BoF / Inverted file	85.532	6.916	292.653	26.152
TSR_{sl} / B-tree	107.218	5.982	235.402	15.319
Δ-TSR_{3D} / B-tree	103.360	6.182	118.618	14.193
Δ-TSR_{5D} / B-tree	115.580	6.232	129.323	13.281
Δ-TSR_{3D} / R-tree	281.721	18.383	328.655	24.343
Δ-TSR_{5D} / R-tree	301.150	19.872	358.323	27.115

Table 2: CPU and IO times (in ms) of the different approaches on the 2 databases DB_{600} and DB_{6000}.

While CPU time is multiplied by two or three for the BoF and TSR_{sl} approaches when using the biggest database DB_{6000}, it is almost constant for Δ-TSR when the index structure is a B-tree. The number of disk accesses is multiplied by 2 for the TSR_{sl} and Δ-TSR, while it increases more for the BoF approach. As a result, these experiments show that the database size has a small influence on Δ-TSR.

To conclude this section, we can say that Δ-TSR really increases the quality of the results (see Section 3) with respectable execution times, compared to the other approaches.

4.2 Strategies for triangle pruning

In this section, we evaluate the relevance of the strategies proposed in section 2.2.2 for pruning triangles, in terms of time retrieval and their consequences on the quality of retrieval with Δ-TSR_{5D}.

Evaluation of the strategies of triangle pruning. Figure 7 and Table 3 respectively present the P/R curves and time retrieval obtained when applying strategy SL, the three variants of strategy SL_{sem} and strategy SL_{geo} on DB_{600}. In this evaluation, these strategies are both applied on the queries and the images of the database.

Method of description		CPU	IO
Δ-TSR_{5D}	- strategy SL	115.580	6.232
	- strategy $SL_{sem+cur}$	71.452	4.561
	- strategy $SL_{sem+bar}$	85.896	4.985
	- strategy $SL_{sem+1st}$	65.637	4.785
	- strategy SL_{geo}	78.714	4.472
BoF		85.532	6.916

Table 3: Average CPU and IO times (ms) for the 3 strategies of pruning with Δ-TSR_{5D} on DB_{600}.

The best results in terms of quality of retrieval are obtained with strategy SL that exploits all the possible triangular relationships in the semi-local neighborhood of each object (see Figure 2 for an example). Such a redundancy in the description of the spatial relationships has the advantage of making the approach more robust to a potential bad repeatability of some points across images. But because of the amount of considered triangle signatures, this strategy is the most expensive in terms of CPU time and IO access. Immediately after, strategy $SL_{sem+cur}$ provides the best quality results: minimizing the amount of triangles in the description

reduces time retrieval drastically, while reducing precision of retrieval only slightly. By considering the center of the neighborhood as pivot, this strategy reduces triangles' stretching, thus improving robustness to viewpoint changes. We observe that with the same complexity, variants $SL_{sem+bar}$ and strategy $SL_{sem+1st}$ provide the worst precision. By considering the objects' barycenter, $SL_{sem+bar}$ minimizes triangles' stretching but is very sensitive to the presence and geometry of any of the objects in the neighborhood, leading to the high risk of determining different pivots in corresponding neighborhoods of two images. Note that CPU time is slightly increased (around 85 ms) because of the barycenter computation in each neighborhood. By choosing the object with the first label as pivot, $SL_{sem+1st}$ may base the triangulation on objects that appear in several neighborhoods, like in the example of Figure 2 with O_1; the disappearance of this object has an impact on the description of the relationships on *several* neighborhoods, while such a configuration is clearly less frequent with $SL_{sem+cur}$. Based on a Delaunay triangulation, SL_{geo} provides precision results slightly worse than $SL_{sem+cur}$, while exploiting more triangles: such a triangulation is *locally* sensitive to the presence/absence and geometry of objects, making it better than $SL_{sem+bar}$ but worse than $SL_{sem+cur}$ that mainly rests on labels. Because strategies SL and $SL_{sem+cur}$ provide the best results, we use them in the following experiments.

Asymmetrical description of the spatial layout. In the previous experiment, the same strategy of pruning was applied both to the image queries and the database. Because in each neighborhood $SL_{sem+cur}$ provides a triangulation which is a subset of the one provided by SL, we also evaluate the scenario where the description of the spatial layout is done *asymmetrically*: query is described using a strategy different to the one used for the database. Table 4 gives the performances obtained in terms of quality and time retrieval (here, because of the paper's size imposed, we only provide the average precision of retrieval).

Strategy		mAP	CPU	IO
Query	DB		(ms)	(ms)
SL	SL	0.743	115.580	6.232
$SL_{sem+cur}$	$SL_{sem+cur}$	0.691	71.452	4.561
$SL_{sem+cur}$	SL	0.721	73.241	5.161
SL	$SL_{sem+cur}$	0.691	79.365	5.585

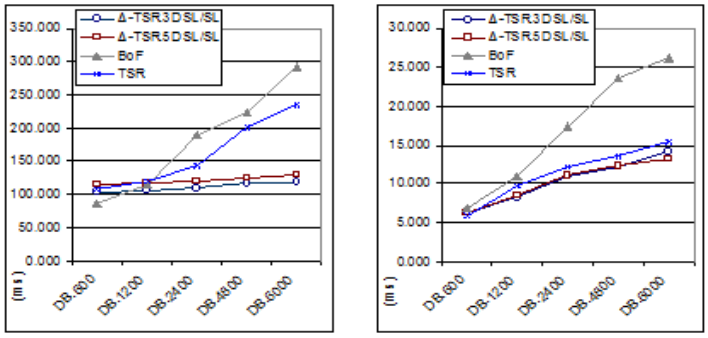
Table 4: Performances of retrieval with Δ-TSR_{5D} and strategies SL and $SL_{sem+cur}$ applied symmetrically and asymmetrically on DB_{600}.

We observe that scenario $SL_{sem+cur}$/SL represents an interesting alternative where quality retrieval remains very close to the best one (SL/SL), while maintaining a low complexity. Then we adopt scenarios SL/SL and $SL_{sem+cur}$/SL for the last experiments that evaluate the scalability of the whole approach by increasing the database size.

4.3 Scalability

In order to evaluate the scalability of Δ-TSR, we vary the size of the database, from 600 up to 6000 images. This evaluation is realized on $N_L = 1000$ labels. In the image retrieval scenario, the main influence on retrieval time concerns the searching technique, the computing of similarity measure and the ranking of resulting images. Figure 8 presents the execution time (CPU and IO) obtained versus the database size for BoF, TSR_{sl} and Δ-TSR with scenario SL/SL.

With BoF and TSR_{sl}, retrieval time increases with the database size, while Δ-TSR_{3D} and Δ-TSR_{5D} present a sub-linear CPU time. Indeed, BoF spends a great part of its execution time on comparing the frequency histograms, therefore greater the visual vocabulary, greater the execution time. Moreover, for a fixed number of labels, the probability of finding images with common labels increases with the database size, and then the number of histograms to compare as well as the volume of inverted file to load from disk. TSR_{sl} associates several



(a) CPU time (b) IO time

Figure 8: Execution time (ms) by varying the database size.

triangles to one signature while Δ-TSR provides one signature per triangle, as already stated in Section 2.1. Consequently, TSR_{sl} presents lower filtering capacity than the Δ-TSR, increasing thus its CPU time because of similarity computation and results ranking. In addition, the complexity of Δ-TSR similarity measure is lower than the one of TSR_{sl}. Another consequence of such codings is that IO times for TSR_{sl} and Δ-TSR are similar – see Figure 8(b), because despite of Δ-TSR high filtering capacity, its B-tree is larger than the one of TSR_{sl}.

Table 5 shows statistics on the volume of features produced for the same databases: thanks to strategy of pruning SL, we see that the number of triangles vary within the same range as the number of points, while it was proved that the triangle-based description is more discriminant. Note that here, the average number of triangles per image is around 500. Consequently, with $N_L = 1000$ (providing 1 billion of possible keys K_1, see Equation 3), in *the most pessimistic configuration* where all the point triplets are different, the current Δ-TSR coding can handle 2 millions of images.

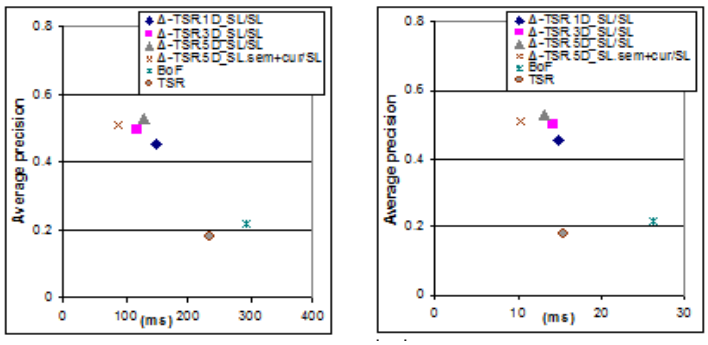
	# of triangles ($r = 8$)		# of points	
Database	in total	per image	in total	per image
DB_{600}	301,091	501	232,486	387
DB_{1200}	570,542	475	453,581	377
DB_{2400}	1,153,687	480	916,764	381
DB_{4800}	2,790,540	581	1,565,701	326
DB_{6000}	2,942,714	490	2,339,807	389

Table 5: Number of triangles and interest points in each database, and their average per image.

To conclude, Figure 9 connects precision and execution times for BoF, TSR_{sl} and the several versions of Δ-TSR on DB_{6000}. We observe that Δ-TSR provides the best ratio precision/execution time, whatever the variant considered.

5 Conclusions and perspectives

This paper is devoted to the representation of spatial relationships between objects in image contents. We have proposed Δ-TSR, which describes triangular spatial relationships with the aim of being invariant to image translation, rotation, scale and robust to viewpoint changes. The approach can be implemented according to 3 variants: $\Delta\text{-TSR}_{1D}$ that describes the co-occurrence of triplets of objects, $\Delta\text{-TSR}_{3D}$ that integrates geometry into $\Delta\text{-TSR}_{1D}$ by describing triplets of objects represented by their centroid, and $\Delta\text{-TSR}_{5D}$ that improves $\Delta\text{-TSR}_{3D}$ by considering the orientation of the objects. In this work, Δ-TSR was evaluated for query-by-example



(a) CPU time (b) IO time

Figure 9: Ratio between mAP and execution time (ms) for several approaches and optimal strategies (SL/SL and $SL_{sem+cur}$/SL) on DB_{6000}.

image and sub-image retrieval by considering low-level visual features called visual words. We demonstrated the relevance of Δ-TSR_{3D} and Δ-TSR_{5D} that improve quality of retrieval notably, compared to two state-of-the-art techniques (a visual vocabulary image representation and TSR). In addition, despite of its theoretical complexity, Δ-TSR was designed to reduce CPU time and IO access notably, by exploiting relevant strategies of triangle pruning and a B-tree as index structure. Its behavior, observed in the last experiments with databases varying from 600 to 6000 images, also demonstrated that this approach is effective and scalable.

Δ-TSR was applied to low-level visual features. Perspectives of this work will consist in considering other kinds of more high-level symbolic objects, and then to propose strategies of pruning different from the current ones that may be not suited because semi-local.

References

[1] G. Bouchard and B. Triggs. Hierarchical part-based visual object categorization. In *CVPR*, pages I 710–715, 2005.

[2] C. Chang. Spatial match retrieval of symbolic pictures. *JISE*, 7(3):405–422, Jan. 1991.

[3] S.-K. Chang and E. Jungert. A spatial knowledge structure for image information systems using symbolic projections. In *ACM Fall Joint Computer Conference, Los Alamitos, CA, USA*, pages 79–86, 1986.

[4] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image Retrieval: Ideas, Influences, and Trends of the New Age. *ACM Computing Surveys*, 40(2):1–60, 2008.

[5] M. J. Egenhofer and R. D. Franzosa. Point set topological relations. *Intl. Journal of GIS*, pages 161–174, 1991.

[6] K. Grauman and T. Darrell. The pyramid match kernel: Discriminative classification with sets of image features. In *IEEE International Conference on Computer Vision*, volume 2, pages 1458–1465, Beijing, China, Oct. 2005.

[7] D. Guru and P. Nagabhushan. Triangular spatial relationship : a new approach for spatial knowledge representation. *Pattern Recogn. Lett.*, 22(9):999–1006, 2001.

[8] P. Huang and C. Lee. Image Database Design Based on 9D-SPA Representation for Spatial Relations. *TKDE*, 16(12):1486–1496, Dec. 2004.

[9] M. Y. Junsong Yuan, Ying Wu. Discovery of Collocation Patterns: from VisualWords to Visual Phrases. In *IEEE*, 2007.

[10] S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In *CVPR*, pages 2169–2178, Washington, DC, USA, 2006.

[11] D. Liu, G. Hua, P. Viola, and T. Chen. Integrated Feature Selection and Higher-order Spatial Feature Extraction for Object Categorization. In *CVPR*, pages 1–8, 2008.

[12] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[13] P. Montesinos, V. Gouet, R. Deriche, and D. Pelé. Matching color uncalibrated images using differential invariants. *IVC Journal*, 18(9):659–672, June 2000.

[14] J. C. Niebles and L. Fei-Fei. A hierarchical model of shape and appearance for human action classification. In *CVPR*, pages 1–8, Minneapolis, MN, 2007.

[15] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8, 2007.

[16] P. Punitha and D. S. Guru. Symbolic image indexing and retrieval by spatial similarity: An approach based on B-tree. *Pattern Recogn.*, 41(6):2068–2085, 2008.

[17] S. Savarese, J. Winn, and A. Criminisi. Discriminative Object Class Models of Appearance and Shape by Correlatons. In *CVPR*, pages 2033–2040, 2006.

[18] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman. Discovering objects and their location in images. In *ICCV*, pages 370– 377, 2005.

[19] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, pages 1470–1477, Oct. 2003.

[20] T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: a survey. *Foundations and Trends in Computer Graphics and Vision*, 3(3):177–280, jan 2008.

[21] W. Wang, Y. Luo, and G. Tang. Object retrieval using configurations of salient regions. In *CIVR*, 2008.

[22] Z. Wu, Q. F. Ke, M. Isard, and J. Sun. Bundling features for large-scale partial-duplicate web image search. In *CVPR*, 2009.

[23] J. Yang, Y.-G. Jiang, A. Hauptmann, and C.-W. Ngo. Evaluating bag-of-visual-words representations in scene classification. In *MIR Workshop*, pages 197–206, 2007.

[24] L. Yang, P. Meer, and D. .Foran. Multiple Class Segmentation Using A Unified Framework over Mean Patches. In *CVPR*, pages 1–8, 2007.

[25] L. Zhu, Y. Chen, X. Ye, and A. Yuille. Structure-perceptron learning of a hierarchical log-linear model. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:1–8, 2008.

[26] L. L. Zhu, Y. Chen, and A. Yuille. Learning a hierarchical deformable template for rapid deformable object parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99(1), 2009.