Content-based Image Retrieval based on Local Affinely Invariant Regions

Tinne Tuytelaars[1] and Luc Van Gool[1,2]

[1] University of Leuven, ESAT-PSI,
Kard. Mercierlaan 94, B-3001 Leuven, Belgium
email: [tinne.tuytelaars, luc.vangool]@esat.kuleuven.ac.be

[2] Kommunikationstechnik, ETH Zentrum, ETZ,
CH-8092 Zürich, Switzerland

Abstract. This contribution develops a new technique for content-based image retrieval. Where most existing image retrieval systems mainly focus on color and color distribution or texture, we classify the images based on local invariants. These features represent the image in a very compact way and allow fast comparison and feature matching with images in the database. Using local features makes the system robust to occlusions and changes in the background. Using invariants makes it robust to changes in viewpoint and illumination.

Here, "similarity" is given a more narrow interpretation than usual in the database retrieval literature, with two images being similar if they represent the same object or scene. Finding such additional images is the subject of quite a few queries.

To be able to deal with large changes in viewpoint, a method to automatically extract local, affinely invariant regions has been developed. As shown by the first experimental results on a database of 100 images, this results in an overall system with very good query results.

1 Introduction

Most existing image retrieval systems mainly focus on color and color distribution. A very popular and well-developed technique is based on color histograms, first developed by Swain and Ballard [11], refined by Funt and Finlayson [2, 3], as well as by Healey and Slater [5] to obtain illumination invariance. Others have tried to use texture (e.g. [7]) or shape (e.g. [4, 9, 1]), but the use of these criteria is mostly limited to a specific application.

We have developed a different approach to content-based image retrieval, based on local invariants. These invariants characterise neighbourhoods of interest points. The latter correspond to corners, found with a corner detector (in our case the Harris detector). The idea is to introduce some geometry in the image retrieval process. Using local features makes the system robust to occlusions and changes in the background. Using invariants makes it robust to changes in viewpoint and illumination. Moreover, the use of invariants allows the use of hashing techniques, resulting in a very efficient image retrieval process.

Probably the most related work is that of Schmid and Mohr [10], who have investigated the matching of points between images based on local greylevel invariants and have also applied this for retrieving images from a database. However, their system only deals with invariance under rotations, combined with a scale space to overcome changes in scale between the images. Therefore, the allowed change in viewpoint is rather limited.

We have extended these ideas towards invariance under more general transformations. More precisely, we consider invariance under affine geometric transformations (valid if the surface over which the invariant is computed is planar) and under linear changes in intensities in each of the three colorbands, i.e. intensities change by a scale factor and offset that may be different for the different color bands (valid in case of lambertian surfaces under changing illumination).

Several invariants satisfying these constraints can be found in literature (e.g. [8]). However, their application in this context is not straightforward. If one wants to use geometric invariants, sets of at least four coplanar points are needed. Apart from the combinatorics, this is unfeasible since there is no way to check the coplanarity of points based on a single image. In case photometric moment invariants are used, one can assume that surfaces are *locally* planar. The problem in that case is that the invariants have to be computed over the same regions in both images, and no finite region can be found that remains unchanged under an affine transformation (for instance, a circular region will be transformed into an elliptical one). One way out would be a generalisation of Schmid and Mohr's scale-space approach. However, with four degrees of freedom for affine invariant regions (e.g. parallelograms), this is computationally unfeasible. Therefore, we first have to find "affinely invariant regions", i.e. we have to develop a method to find the same region in both images independently.

In section 2, we describe a method to find affinely invariant regions in order to handle the geometric distortions between the query image and a corresponding image in the database. Invariant-based image retrieval using these invariant regions is discussed in section 3. These invariants also cover for the photometric changes between views. Section 4 shows some experimental results and section 5 concludes the paper with some final remarks.

2 Finding Affinely Invariant Regions

The problem addressed in this section can be summarized as follows: given two images of the same scene, taken from different viewpoints, find one or more regions around a point, such that the same region(s) is (are) found in both images *independently*, i.e. without using knowledge about the other image.

In a different context, Lindeberg et al. [6] have also been looking into this problem. Their approach is situated in the domain of shape from texture, where they apply "affine shape adaptation" of smoothing kernels. In the case of weak isotropy, the region found corresponds to rotationally symmetric smoothing and rotationally symmetric window functions in the tangent plane to the surface. However, for other cases, their method does not necessarily converge.

We consider invariance under affine geometric and photometric changes (i.e. a scale and offset for each spectral band). Thus, we assume that the scene is locally planar and not occluded and that no strong, specular reflections occur.

To reduce the complexity of the problem, we restrict ourselves to finding affinely invariant regions for corner points making use of the nearby edges. Also, we make a distinction between two different cases: curved edges and straight edges.

2.1 Case 1: Curved Edges

Let $\mathbf{p} = (x_p, y_p)$ be a corner point, and e_1 and e_2 the edges in its neighbourhood. Then two relative affinely invariant parameters l_1 and l_2 can be defined for the two edges e_1 and e_2, in the following way (see also fig. 1)

$$l_i = \int abs(|\mathbf{p_i}^{(1)}(s_i) \quad \mathbf{p} - \mathbf{p_i}(s_i)|)ds_i \qquad i = 1,2$$

with s_i an arbitrary curve parameter, $\mathbf{p}^{(1)}(s_i)$ the first derivative of $\mathbf{p_i}(s_i)$ with respect to s_i, $abs()$ the absolute value and $|..|$ the determinant. Then, a point $\mathbf{p_1}(l_1)$ on one edge can be associated with a point $\mathbf{p_2}(l_2)$ on the other edge, such that $l_1 = l_2$. Both l_1 and l_2 are relative, affine invariants, but their ratio $\frac{l_1}{l_2}$ is an absolute affine invariant and the association of a point on one edge with a point on the other edge is also affinely invariant. From now on, we will simply use l when referring to $l_1 = l_2$.

Together, the two points $\mathbf{p_1}$ and $\mathbf{p_2}$ define a region A for the point $\mathbf{p}$ as a function of l: the parallelogram spanned by the vectors $\mathbf{p_1} - \mathbf{p}$ and $\mathbf{p_2} - \mathbf{p}$. In this way, the problem of finding an affinely invariant region has been reduced to finding a value for l in an affinely invariant way.

To this end, we evaluate a function over the region $A(l)$ that reaches its extrema for corresponding values of l, i.e. in an invariant way for both the geometric and photometric changes. We then select the region $A(l)$ for which such function reaches a local extremum. Since it is not guaranteed that the function will really reach an extremum over the limited l-interval we are looking

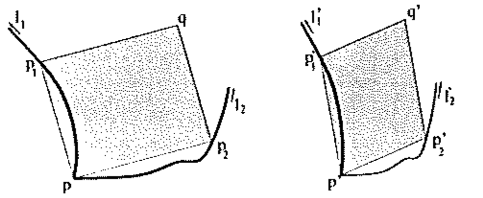

Fig. 1. Based on the edges in the neighbourhood of a corner point, an affinely invariant region can be found as a function of the relative affinely invariant parameter $l_1 = l_2$.

at, a set of functions are tested. As a result, more than one region might be found for one corner point. The functions we used in our experiments are:

$$\frac{\int I(x,y)\ dxdy}{\int dxdy} \quad and \quad \frac{|\mathbf{p}-\mathbf{q} \quad \mathbf{p}-\mathbf{p_g}|}{|\mathbf{p}-\mathbf{p_1} \quad \mathbf{p}-\mathbf{p_2}|}$$

with $I(x,y)$ the image intensity, $\mathbf{p_g}$ the center of gravity of the region, weighted with image intensity, and $\mathbf{q}$ the corner of the parallelogram opposite to the cornerpoint $\mathbf{p}$ (see figure 1).

$$\mathbf{p_g} = \left(\frac{\int I(x,y)x dxdy}{\int I(x,y) dxdy}, \frac{\int I(x,y)y dxdy}{\int I(x,y) dxdy}\right)$$

2.2 Case 2: Straight Edges

In the case of straight edges, the method described above cannot be applied, since the relative affinely invariant parameter l will be zero. However, since straight edges occur quite often in real images, we cannot just neglect this case.

A straightforward extension of the previous technique would then be to search for local extrema of some appropriate function in a two-dimensional search-space with the two parameters running over the edges as coordinates, instead of a one-dimensional search-space running over l.

However, the functions which we have been using do not show a clear, well-defined extremum. Instead, we have some shallow "valleys" of low values, (e.g. corresponding to cases where the area in the numerator tends to zero). Instead of taking the (inaccurate) local extremum, we combine two such functions, and take the intersection of the two valleys, as shown in figure 2. Of course, it is not guaranteed that the valleys will indeed intersect. That is why different combinations of invariants are tested (e.g. for different subregions, or for different color bands). Moreover, the special case where two valleys (almost) coincide must be detected and rejected, since the intersection will not be accurate in that case.

3 Image Retrieval based on local invariants

Now that we are able to extract affinely invariant regions in an image, local, affine moment invariants can be computed and used to match a query image

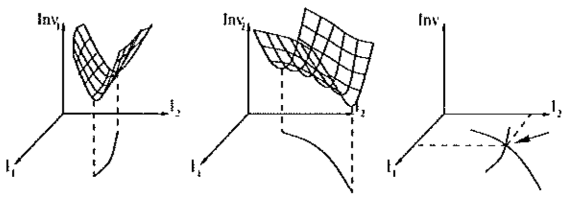

Fig. 2. For the straight edges case, the intersection of the "valleys" of two different invariants is used instead of a local extremum.

with images in the database representing the same scene or object. First, we describe what local invariant features we use. Next, the image retrieval process is discussed in more detail.

3.1 Local Invariant Features

As in the region finding step, we consider invariance both under affine geometric changes and linear photometric changes, with different offsets and scalefactors for each of the three color bands.

For each region, a feature vector is composed, consisting of affine moment invariants. These can be matched quite efficiently with the invariant vectors computed for the regions in the database images, using a hashing-technique. In this way, combinatorics can be avoided, reducing the computation time from $O(nN)$ to $O(n)$ (with n the number of regions in the query image and N the total number of regions in all of the database images). As a result, the processing time is independent of the size of the database.

As a first invariant– and, as shown by our experiments, a quite distinctive one – we use the "type" of the region found. Here, with "type" we refer to the method used to find the region: was it found using the method for curved edges or with the method for straight edges ? and what functions were used ? Only if the type of two regions corresponds, can they be matched.

The other invariants are all affine moment invariants. Care has to be taken, however, that they are sufficiently distinctive. More specifically, one must pay attention to the fact that the region finding process turns some invariants into trivial cases.

Due to space limits, a detailed description of the invariants used in our experiments can not be given here. Therefore, we refer to [8]. Briefly, moment invariants up to the second order are computed, that use each of the three color bands to obtain a higher distinctive power.

3.2 Image Retrieval based on Local Invariant Features

In order to find the corresponding image(s) in the database, a voting mechanism is applied. Each point in the query image is matched to the point in one of the database images whose feature vector yields the smallest Mahalanobis-distance. If this distance is smaller than some threshold th, the cross correlation between them is computed (after normalization to a reference region) as a final check to reject false matches. Each match between a point in the query image and a point in one of the database images is then translated into a vote for the corresponding database image. The image that receives the highest number of votes, is selected as the query result.

From the ratio between the highest number of votes and the second highest number of votes we can derive a "confidence-measure" c:

$$c = \frac{n_1}{n_1 + n_2} \qquad if \quad n_2 < n_1$$

$$c = \frac{1}{N} \qquad if \quad n_2 = n_1$$

with n_1 the highest number of votes, n_2 the second highest number of votes and N the number of images with n_1 votes. The higher this value, the more confident we are that the query result is correct. Indeed, if the second highest number of votes is zero, the confidence measure is estimated to be 100 %. If, on the other hand, the second highest number of votes is equal to the highest number of votes, the estimated confidence measure is at most 50 % and depends on the number of images with the same number of votes.

4 Experimental results

For our experiments, we used sets of two or exceptionally three color images of a scene or object. Each time, one image was kept as a query image, while the other(s) was (were) put in a database which, at the end, contained 50 images of more than 40 different scenes or objects. To this, we added 50 more images from the Corel Professional Photos database that were selected from the packages "Big Apple", "Egypt", "India" and "China", to end up with a total of 100 images.

Fig. 3 shows some query images (upper rows) together with the image of the same object or scene in the database (lower rows). Note that there typically is a large difference in viewpoint between both images. Also the illumination conditions may have changed, and parts of the object or scene may have been occluded in one of the images. Nevertheless, the correct image was retrieved in almost 70 % of the query images as the image with the highest number of votes. In 95 % of the cases, the correct image was among the upper seven retrieved images. Examples of images where the system failed are the ones shown in the most right column of fig. 3. In the case of the car-images, this is due to the lack of texture on the car, combined with specular reflections. For the lower right images, the scene is composed of many different objects, causing many occlusions that mislead the matching process. For all the other images in fig. 3 the correct image was the first one retrieved. Table 1 gives an overview of the results.

The ranking as obtained through the voting process must be interpreted as a ranking according to the probability of representing the same object or scene. In contrast to other content-based image retrieval systems, our system cannot be used to rank the database images according to their similarity with the query image. This is caused by the fact that only local similarity is checked, while a human observer usually looks at similarity on a global scale. However, it is our belief that in many applications what the user is really interested in are images with the same objects or scenes rather than a more qualitative similarity.

5 Summary and Conclusions

In this contribution, a novel approach to content-based image retrieval was worked out, based on local, illumination and viewpoint invariant features. To

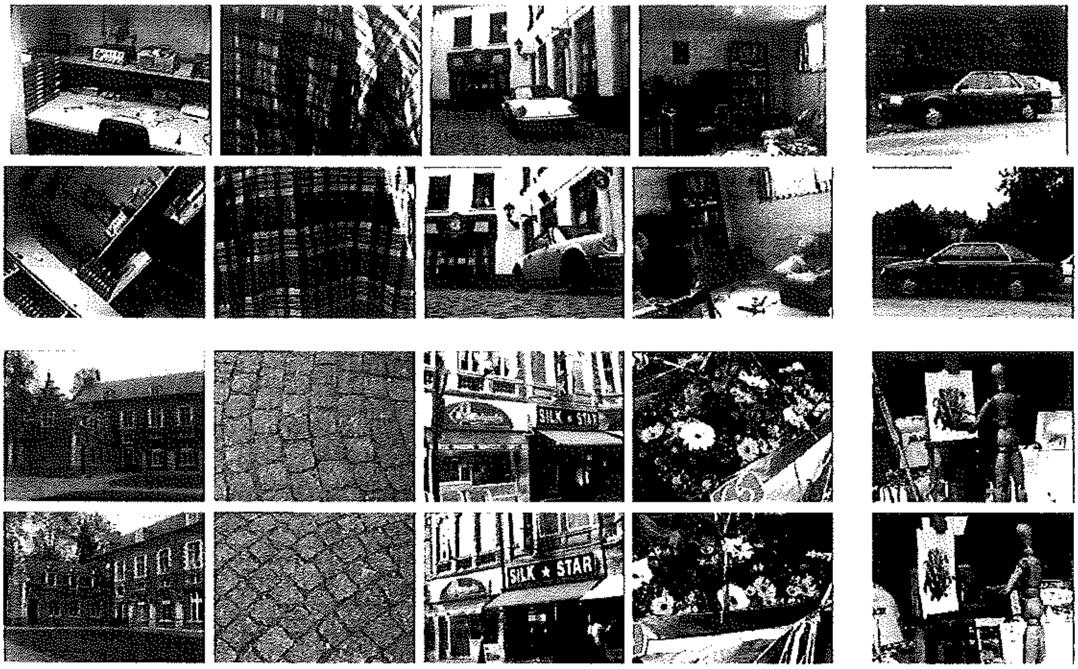

Fig. 3. Some examples of query images (upper rows) and their corresponding images in the database (lower rows).

this end, a method for the automatic extraction of affinely invariant regions was developed.

The first results on a database of 100 images showed that the method is promising, and can be considered a good alternative to the existing, color or texture based retrieval systems in cases where images of the same object or scene are searched for. Of course, further testing is still required, both on a larger database and using more complex criteria (based on combinations of our approach and global, color-based techniques, for example).

Acknowledgments

TT is a Research Assistant of the Fund for Scientific Research (FWO). Support by IUAP project 'Intelligent Mechatronic Systems' of the Belgian DWTC and by the Fund for Collective Fundamental Research (FKFO) of the Fund for Scientific Research (FWO) is gratefully acknowledged.

References

1. A. Del Bimbo and P. Pala, *Effective Image Retrieval Using Deformable Templates*, ICPR96, pp. 120 - 123, 1996.
2. G. D. Finlayson, S. S. Chatterjee and B. V. Funt, *Color Angular Indexing*, ECCV96, Vol.II, pp.16-27, 1996.
3. B. V. Funt and G. D. Finlayson, *Color Constant Color Indexing*, IEEE PAMI Vol. 17, no. 5, pp.522-529, 1995.

Table 1. Image Retrieval Results: For each image (first column), the position at which the correct image from the database was retrieved (second column) together with the confidence ratio (third column) is given.

image	position of correct image	confidence measure
auto	7	
bak	3-5	
bank	1	0.881
bellavista	1	0.706
bier	1-3	0.333
bloemen	1	0.706
choc	1	0.9
cola	1-4	0.25
deur	6	
domus	1	0.65
heroon	3-5	
kassei	1	0.538
kasteel	1	0.684
kobe	4-6	
kot	6-10	
kotp	1	0.545
mask	2-5	
masker	14	
matras	1	0.929
meter	5-18	
display	3-4	

image	position of correct image	confidence measure
model	11-19	
plant	1	0.75
poort	5	
post	1	0.71
rob	1	0.636
simpsons	1-4	0.25
stof	1	0.805
tank	1	0.611
tankf	1	0.905
tegels	1	0.6
winkels	1	0.739
boor	2-10	
pc	1	0.577
muur	1	0.666
koffie	1	0.6
stapel	1	0.565
trio	1	0.848
ikke	1	0.8
tractor	1-2	0.5
dak	1	0.947

4. E. Gary and R. Mehrorta, *Shape similarity-based retrieval in images databases*, SPIE, Vol. 1662, 1992.
5. G. Healey and D. Slater, *Global Color Constancy: Recognition of Objects by Use of Illumination Invariant Properties of Color Distributions*, J. Opt. Soc. Am. A, Vol. 11, no. 11, pp. 3003-3010, nov 1995.
6. J. Gårding and T. Lindeberg *Direct computation of shape cues using scale-adapted spatial derivative operators*, IJCV, Vol. 17, no. 2, pp. 163-191, feb 1996.
7. F. Liu and R. W. Picard, *Periodicity, directionality, and randomness: Wold features for image modeling and retrieval*, M.I.T. Media Laboratory Perceptual Computing Section Technical Report, No. 320, 1995.
8. F. Mindru, T. Moons, L. Van Gool *Color-based Moment Invariants for the Viewpoint and Illumination Independent Recognition of Planar Color Patterns*, to appear at ICAPR, Plymouth, 1998.
9. F. Mokhtarian, S. Abbasi and J. Kittler, *Robust and Efficient Shape Indexing through Curvature Scale Space*, BMVC(96), September 9-12, 1996.
10. C. Schmid, R. Mohr *Local Greyvalue Invariants for Image Retrieval*, PAMI Vol. 19, no. 5, pp 872-877, may 1997.
11. M. J. Swain and D. H. Ballard, *Color Indexing*, IJCV, Vol. 7, no. 1, pp. 11-32, 1991.

This article was processed using the LaTeX macro package with LLNCS style