RETRIEVING OBJECTS FROM VIDEOS BASED ON AFFINE REGIONS

Vittorio Ferrari[1], *Tinne Tuytelaars*[2], *Luc Van Gool*[1,2]

[1]Computer Vision Group (BIWI), ETH Zuerich, Switzerland
[2]ESAT-PSI, University of Leuven, Belgium
{ferrari,vangool}@vision.ee.ethz.ch, Tinne.Tuytelaars@esat.kuleuven.ac.be

ABSTRACT

We present a method to (semi-)automatically annotate video material. More precisely, we focus on recognizing specific objects and scenes in keyframes. Objects are learnt simply by having the user delineate them in one (or a few) images. The basic building block to achieve this goal consists of affine invariant regions. These are local image patches that adapt their shape based on the image content so as to be invariant to viewpoint changes. Instead of simply matching the regions and counting the number of matches, we propose to gather more evidence about the presence of the object by exploring the image around the initial matches. This boosts the performance, especially under difficult, real-world imaging conditions. Experimental results on news broadcast data demonstrate the viability of the approach.

1. INTRODUCTION

We tackle the problem of automatically finding a specific object or scene in a video. The object is only given as delineated by the user in one, or a few, model images. This is useful in the context of video annotation: instead of having a person manually marking every occurrence of interesting objects in the video, the system could do this largely automatically, resorting to human intervention only to delineate each object once.

Traditionally, in such content-based image retrieval task the object is represented by *global* features, which collect information across the whole image. Examples of such features are histograms of color and texture, or edge-signatures [10].

In the last few years, several techniques for extracting local, invariant features have appeared [11, 6, 7, 1, 5]. The features are small, planar image regions whose extraction process is invariant to affine geometric transformations. Thus, the regions are detected *independently* in each image, while their shape automatically adapts to the viewpoint so as to keep on covering the same physical surface (figure 1).

Typically, retrieval schemes characterize the regions by a vector of invariant descriptors and use them to match model regions to regions coming from a test image (e.g.: a video frame) in a nearest-neighbor style. Images with a sufficient amount of matches are labeled as containing the object.

Unlike with global methods, representing the object as a collection of local regions brings robustness to background clutter and partial occlusions. Moreover, viewpoint changes are allowed and the search focuses on a specific object, rather than on mere similar appearance (e.g.: color histogram).

The authors gracefully acknowledge support from EU projects CIMWOS and VIBES, the Fund for Scientific Research Flanders.

Figure 1: *Example case. A pair of corresponding regions independently extracted from a model-image (left, only user-delineated parts shown) and a keyframe (right).*

Unfortunately, the chance of repeatably extracting a given region in both the model-image and a test video frame considerably drops under a combination of challenging conditions, such as large scale and viewpoint changes, strong occlusion and the low image quality a video usually offers. At the same time, extensive clutter generates many spurious features, which disturb the matching process. As a final outcome, only a few regions are correctly matched, which is insufficient for reliable object detection in a long video.

In this paper, we tackle the problem by no longer relying solely on matching viewpoint invariant regions. Instead, we anchor on an initial set thereof, and then *look around* them trying to construct more matching regions. As new matches arise, they are exploited to construct even more, in a process which gradually explores the video frame, recursively constructing more and more matches, increasingly further from the initial ones. This *expansion* process is alternated with a *contraction* one, where incorrect matches are removed. As the number and extent of matching regions grows, so does the system's confidence in the presence of the object.

The basic characteristic of the approach is that each single correct initial match can expand to cover a contiguous surface of the object with many correct matches, even when starting from a majority of mismatches. This leads to several advantages. First, robustness to scale, viewpoint, occlusion and clutter are greatly enhanced, because most cases where the original approach produced only a few correct matches can now be solved. Second, the approximate boundaries of the object are directly indicated by the envelope of the final set of matches. Recognition and segmentation are achieved at the same time. Third, non-rigid deformations are taken into account. The method can extend any viewpoint invariant region extractor.

In [9] another region-based system for video object retrieval is presented. However, it focuses on different aspects of the problem, namely the organization of regions coming from several shots, and weighting their individual relevance in the wider context of the video. At the feature level, their

work still relies solely on regions from standard extractors.

This paper is organized as follows. The next section gives a scheme of the system. Section 3 explains the image exploration algorithm, while experimental results on news broadcast material are reported in section 4.

2. SCHEME OF THE SYSTEM

The system's input consists of a model-image containing the object to be annotated delineated by the user, and a test video stream where to search for the object. The model-image does not necessarily have to come from the test video.

The processing is divided in learning and recognition. During learning, regions are extracted from the object part of the model-image. We use [11], but any affine invariant region extractor is suitable.

The recognition phase goes through the following stages:

1. *Video segmentation.* The input video is segmented into shots, and a few representative keyframes are selected in each shot. The video is sampled so that subsequent keyframes are significantly different, but still adequately cover the whole content of the shot. This operation is performed by the algorithm of [8]. Further stages only inspect the keyframes.
2. *Region extraction.* Regions are extracted in all keyframes, with the algorithm of [11].
3. *Keyframes exploration.* The regions of a keyframe are first matched to the model regions. The surrounding area is then gradually explored. The process tries to cover the whole object with new matching regions, while simultaneously removing mismatches.
   This stage is the subject of the paper, and is described in more detail in the next section.
4. *Detection.* The object is detected in every keyframe with more than a pre-defined amount of matches (after the exploration stage).

3. EXPLORING THE KEYFRAMES

The case of figure 1 poses several challenges. The object is physically bent and it appears smaller and occluded in a heavily cluttered keyframe. The images, which come from a compressed video of a news broadcast, are of low quality.

Because of these difficult conditions, the approach of [11] produces only 10 matches, out of which 5 are correct. Although this figure somehow indicates a detection, it is not sufficiently high to guarantee good performance in a long video. Indeed, also many keyframes which *do not* contain the object might obtain 10, or more, matches. Besides, for reliability it would be preferable to identify a larger percentage of correct matches than 50%.

In this section, we present a method [1] which bootstraps from the initial matches produced by [11] and, in case the object is present in the keyframe, generates *many* more. This results in much higher detection scores when the object is present than when it is not.

3.1 Coverage of the model image

The model image is densely covered with a grid of overlapping circular regions. At this point, none of them is matched to the keyframe. The goal of the following steps is to generate in the keyframe as many regions corresponding to these as possible.

[1]This is a simplified version of our method [3] adapted for the application. It was first used for object recognition in still images.

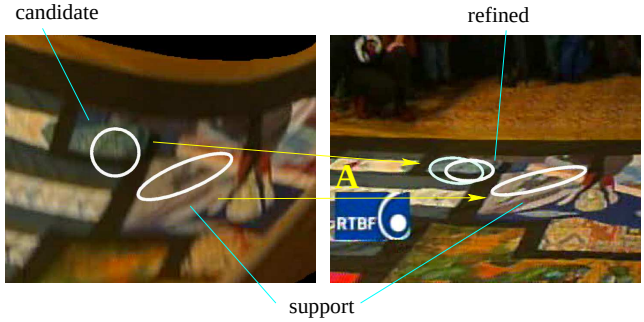


Figure 2: *A candidate region is propagated via the affine transformation of a support match. Refinement adapts the shape to the different surface orientation of the candidate.*

3.2 Expansion

The geometric transformation of the current matches are used to construct new keyframe regions which match nearby coverage regions. The procedure is illustrated in figure 2. Let **A** be the geometric (affine) transformation mapping an existing region match R from the model image to the keyframe. **A** is applied to an unmatched coverage region lying near R in the model image. If both the *support* region R and the *candidate* coverage region lie on the same continuous physical surface, then they will be mapped to the keyframe by similar affine transformations. Hence, this operation constructs a region in the keyframe roughly corresponding to the candidate.

The geometric registration is refined by the algorithm of [2], which modifies the shape of the newly created region so as to maximize the similarity with the candidate in the model image. A combination of greylevel normalized-cross correlation and pixel-wise distance in normalized RGB color space is used as similarity measure. This refinement step adapts the region to the local surface orientation and to perspective effects. The refined region is accepted as match for the candidate if its final similarity exceeds a threshold. We say that the candidate is *propagated* to the keyframe. When several supports are present around a candidate, only the best propagation is kept.

The propagation scheme is applied to all currently unmatched coverage regions. As a total effect, many new matches are generated, in an area around the previous matches. However, not all of them are guaranteed correct. Some *mispropagations* are possible, for example when a propagation based on a mismatched support is accepted.

3.3 Contraction

In order remove incorrect matches, the current matches are filtered based on the analysis of their spatial arrangements. The filter is based on the following *sidedness constraint* for a triple of region matches. The center of a first region should be on the same side of the directed line going from the center of a second region to the center of a third region, in both the model-image and the keyframe. This constraint holds for the large majority of correctly matched triples (see [2] for details).

A triple including any mismatched region has higher

Figure 3: *The contour of the final set of 130 matches. Note the completeness of coverage and the accuracy in general.*

chances to violate the constraint. When this happens, we can only conclude that probably at least one of the matches is incorrect, but we do not yet know which one. However, by integrating the weak information each triple provides, it is possible to robustly discover mismatches. We check the constraint for all unordered triples and we expect wrong matches to be involved in a higher share of violations.

The filter algorithm starts from the current set of matches, and then iteratively removes matches as follows:

1. (Re-)check the constraint for all triples of current matches. For each match, store the percentage of constraints it violates.
2. Find the worst match W, which violates most constraints.
3. If the percentage of violations of W is above a threshold, remove W from the set of matches, and iterate to point 1, else stop.

The idea of the algorithm is that at each iteration the most probable mismatch W is removed and the number of violations for correct matches decreases, because they are involved in less triples containing any mismatch. After several iterations, ideally only correct matches are left. Since these are involved in only a few violations, the algorithm stops.

The proposed filter has two main advantages over detecting outliers to the epipolar geometry through RANSAC [4], which is traditionally used in the matching literature. It allows for non-rigid deformations and it is much less sensitive to inaccurate localizations of the regions.

3.4 Alternate expansion and contraction

The processing continues by iteratively alternating expansion and contraction phases:

1. Do an expansion phase. All current matches are used as supports, and all original coverage regions that are not yet matched are candidates. The propagated region matches are added to the current matches.
2. Do a contraction phase on the current set of matches.
3. If at least one newly propagated region survives the contraction, then iterate to point 1. Otherwise stop.

In the first iteration, the expansion phase generates many correct matches, along with some mismatches. The first contraction phase removes mostly mismatches, but might also lose some correct matches: the percentage of wrong matches might be still high and confuse the filter. In the next iteration, this cleaner configuration is fed into the expansion phase which, less distracted, generates more correct matches and less mismatches. The presence of new correct matches in turn helps the next contraction in taking better removal decisions, and so on.

As a result, the amount and spatial extent of the correct matches grow at every iteration, reinforcing the confidence about the object's presence and location. The two processes of expansion and contraction *cooperate* in order to gather more evidence about the object and separate correct matches *at the same time*.

Thanks to the refinement, each expansion phase adapts the shape of newly created regions to the local surface orientation. Thus the whole exploration follows curved surfaces and deformations.

The approach fills the visible portion of the object with many high confidence matches. Besides bringing discriminative power, this results in the simultaneous recognition and segmentation of the object.

Applying this scheme to the example of figure 1 yields 130 final matches, 117 of which are correct (figure 3). This is much better than the 10 matches of [11], the approach we started from. Moreover, the exploration procedure tends to 'implode' when the object is not in the keyframe, typically returning 0, or at most a few, matches. Hence, discriminative power is greatly increased when searching for many objects in a long video.

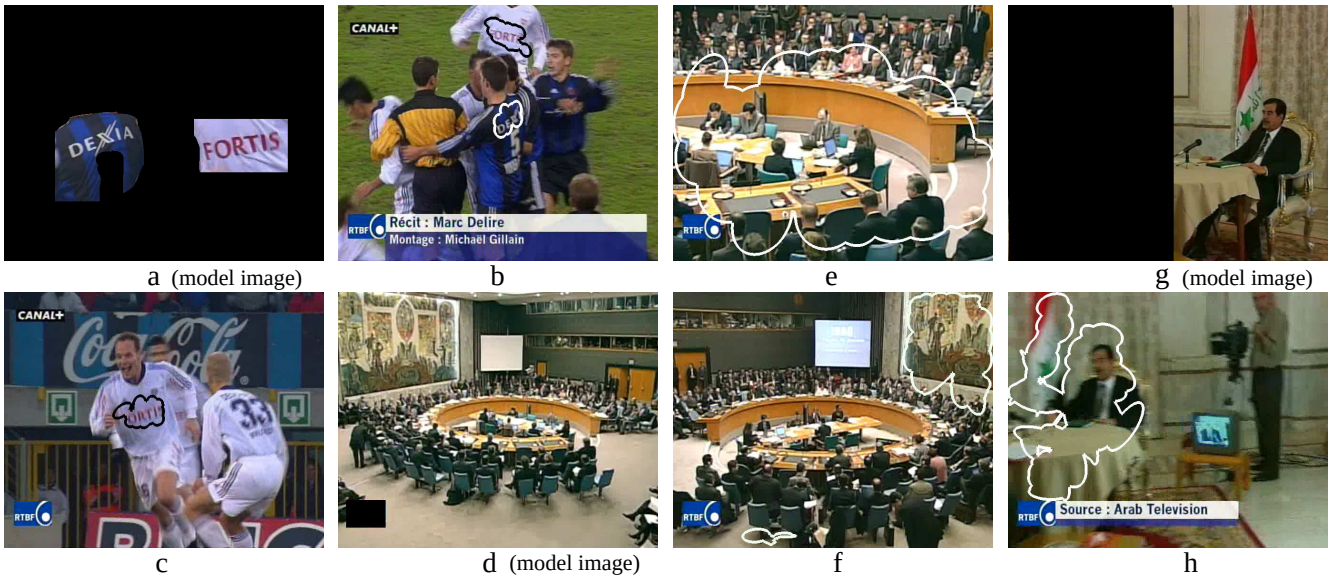
4. RESULTS AND CONCLUSIONS

We report results for news broadcast material from the RTBF Belgian television channel. The data comes from 4 different news report videos, captured on different days, each of about 20 minutes. Keyframes were obtained through the algorithm of [8]. The image quality is quite low: the keyframes have low resolution (672x528) and many of them are visibly affected by compression artifacts, motion blur and interlacing effects. We selected 13 objects, including locations, advertising products, logos and football shirts, and delineated each in one representative keyframe. Each object is searched in the keyframes of the video containing its model-image. In average, an object is searched in 325 keyframes, and occurs 7.4 times. The number of 'negatives', i.e.: keyframes not containing an object, is therefore much greater than the number of positives, which allows to collect significant statistics. A total of 4236 (object,keyframe) image pairs have been processed.

Figures 3 and 4 show some examples of successful detections. A large piece of cloth decorated with various flags is found in figure 3 in spite of non-rigid deformation, occlusion and extensive clutter. Notice the completeness of the segmentation (the right part is self-occluded in the model and, correctly, left undetected in the keyframe, as is the part occluded by the RTBF logo).

An interesting application is depicted in figures 4a-b-c. The shirts of two football teams are picked out as query objects (figure 4a) The system is then asked to find the keyframes where the first team (Dexia) is playing, and where the other team (Fortis) is playing. In figure 4c the Fortis shirt is successfully found in spite of moderate rotation and motion blur. Both teams are identified in figure 4b, even if the shirts appear much smaller and the Dexia player is turned 45 degrees (viewpoint change on the shirt).

Robustness to large scale changes and occlusion is demonstrated in figure 4e, where the UN council, modeled in figure 4d, is recognized while enlarged by a scale factor 2.7, and heavily occluded: only 10% of the model image is visible. Equally intriguing is the image of figure 4f, where



a (model image) b e g (model image)

c d (model image) f h

Figure 4: *Results. The parts of the model-images not delineated by the user are blanked out. The sizes have not been altered: the model images are shown at the same scale as the test keyframes. Details in the text.*

the UN council is seen from an opposite viewpoint. The large painting on the left in the model image is about the only thing still visible in the test keyframe, where it appears on the right side. The system managed to match the whole area of the painting, which suffers from out-of-plane rotation, and thus retrieve the UN council.

As a last example, a room with Saddam Hussein is found in figure 4h (model in 4g). The keyframe is taken under a different viewpoint and corrupted by considerable motion blur.

The retrieval performance is measured by the *detection rate* and *false positive rate*, averaged over all 13 objects. For an object, the detection rate is the number of correct detections divided by the total number of times the object occurs in the video (to keep results fair, detections of model-keyframes are not counted). The false positive rate refers to the number of wrong detections over the number of negatives. An object is detected if the number of final matches, divided by the number of model coverage regions (see subsection 3.1), exceeds 10%. The system performs well, by achieving an average detection rate of 82.4%, for a false-positive rate of 3.6%. As a comparison, we repeated the whole experiment with [11], the method we started from. It only managed a 33.3% detection rate, for a false-positive rate of 4.6%, showing that our approach can substantially boost the performance of standard affine invariant matching schemes.

It takes on average 2.16 minutes to process a (object,keyframe) pair on a modest workstation (1.4 Ghz PC). While this is not particularly fast, and far slower than real-time, it is still a reasonable computational requirement for off-line processing. In this scenario, the system is run beforehand on many potentially interesting objects, and user-queries are processed in real-time based on the pre-computed annotations (like in [9]).

The results confirm the viability of our approach for retrieving objects in the challenging, real-world conditions of news broadcast video data. The method is very effective against viewpoint and scale changes, occlusion, clutter and is robust to moderate amounts of image degradation, like motion blur and compression artifacts. Moreover, deformable objects are taken into account and the approximate contours of the object are produced.

Potential improvements include the support of uniformly colored, or very sparsely textured, objects, the reduction of computational requirements, and the exploitation of the video's temporal continuity (e.g.: for learning a multi-pose model from various sides of an object visible in the shot surrounding the model keyframe). .

REFERENCES

[1] A. Baumberg, Reliable feature matching across widely separated views *Intl. Conf. on Comp. Vis.*, 2000.

[2] V. Ferrari, T. Tuytelaars and L. Van Gool Wide-baseline Multiple-view Correspondences, *IEEE Comp. Vis. and Patt. Rec.*, 2003.

[3] V. Ferrari, T. Tuytelaars and L. Van Gool Simultaneous Object Recognition and Segmentation by Image Exploration, *European Conf. on Comp. Vis.*, 2004.

[4] M.A. Fischler and R.C. Bolles, Random Sampling Consensus - a paradigm for model fitting with applications to image analysis and automated cartography, *Commun. Assoc. Comp. Mach.*, 1981.

[5] D. Lowe, Distinctive Image Features from Scale-Invariant Keypoints *submitted to IJCV*, 2004.

[6] K. Mikolajczyk and C.Schmid, An affine invariant interest point detector *European Conf. on Comp. Vis.*, vol. 1, 128-142, 2002.

[7] S. Obdzalek and J. Matas, Object Recognition using Local Affine Frames on Distinguished Regions *British Machine Vision Conf.*, pp. 414-431, 2002.

[8] M. Osian and L. Van Gool, Video Shot Characterization, *TRECVID workshop*, 2003.

[9] J. Sivic and A. Zisserman, Video Google: A Text Retrieval Approach to Object Matching in Videos *ICCV*, 2003.

[10] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, Content-based Image Retrieval at the End of the Early Years *PAMI*, 22(12), pp. 1349-1380, 2000.

[11] L. Van Gool, T. Tuytelaars, A. Turina, Local features for image retrieval, *in State of the art in content-based image and video retrieval.*, eds. R. C. Veltkamp et al., pp. 21-42, 2001