Computer Aided (dis)Assembly Using Visual Cues

Bart Lamiroy, Cordelia Schmid, Roger Mohr, Martin Tonko, Kart Schäfer, Hans H. Nagel

Computer Aided (dis)Assembly Using Visual Cues.*

B. Lamiroy, C. Schmid, R. Mohr

MOVI
GRAVIR Laboratory†

655 Avenue de l'Europe
38330 Montbonnot St. Martin, France

M. Tonko, K. Schäfer, H.-H. Nagel

Institut für Algorithmen
und Kognitive Systeme
University of Karlsruhe
Am Fasanengarten 5, Postfach 6980
76128 Karlsruhe, Germany.

Abstract

In this paper we propose a new vision based method for realizing automated disassembly tasks. We applied our method to identification and localization of parts of a car engine, but the method can be generalized to a broad range of assembly or disassembly tasks. The general outline of the algorithm we propose is the following :

1. *Identifying the global object and its pose in correspondence with known models .*
2. *Extrapolating the visual information to localize individual parts in the scene.*
3. *Visually control the tasks concerning the found parts.*

We shall mainly focus on the first two steps of the given algorithm, referring to the last step only briefly.

The originality of our approach mainly lies in two points :

- *The "models" used for the recognition task consist of selected views of the objects to recognize (in our case, pictures of car engines) obtained by a standard image acquisition system. This allows us to have a robust recognition algorithm, without any influence of synthetic information which could haphazard identification.*
- *In order to "extrapolate" the synthetic information of a higher semantic nature, which is necessary for the assembly/disassembly task, the model images are enriched with the needed information on the right spots. In other terms, symbolic information is added to the model images. Use of the trilinearity constraint, for instance, then allows us to locate precisely the parts that need to be accessed, even if they are invisible in the control images.*

1 Introduction

One of the promising and interesting applications of computer vision is visual control. Indeed, numerous industrial tasks or processes would be greatly improved if a visual control process was integrated into the global control loop. Major impacts would be found in the domain of flexibility, reduced calibration times, or less stringent positioning requirements, if the system were able to identify in a (semi) autonomous manner the objects it is to manipulate, and to find their location in the observed scene. In order to accomplish this kind of tasks, it needs certain amount of information concerning these objects in order to identify and manipulate them in a correct way.

Several authors [10], [2], [1] have shown that use of synthetic data for modeling objects can have a very negative influence on object recognition tasks. Chen and Mulgaonkar [1] have shown, that for simple objects, the most sophisticated and complete synthetic models could not predict with sufficient accuracy the aspect of the real image.

On the other hand, the information that can be embedded in synthetic data is of a capital importance for automatically manipulating objects in a wide range of applications (industrial assembly/disassembly being one of them [11]), one could for instance consider density, weight, type of matter, coefficients of friction

*This paper was presented at the I.A.R. Annual Meeting, Karlsruhe, Germany, November 1996

†A joint research project between CNRS, INPG, INRIA and UJF.

or resistance to pressure, etc. For sake of convenience, we shall refer to the totality of supplementary data, be it synthetic or obtained through measurements, as CAD data.

It is clear that use of CAD only is insufficient for obtaining valid results in a real world environment where recognition and localization of objects is concerned. It is clear also, that mere use of visual data cannot pretend to offer the physical information that an industrial application would require. We therefore present a method integrating real visual data as well as CAD information using either where necessary. We define an environment of application, which we model by a number of key views, taken with standard imagery equipment. According to the tasks we need to perform, we enrich these images with projections of CAD models of the objects we need to manipulate in the scene, at the spots they are located at in the image. When an unknown image of a scene is shown to our system, we use only the visual information to get a positioning relative to the appropriate key views. Once this position obtained, we are capable of extrapolating the known CAD objects in the new scene. In our application, the model images consist of overall views of different car engines, taken from several viewpoints; the CAD objects are individual parts within the engine.

The outline of this paper is the following : we first give a brief recall of the recognition algorithms and their justification used in our approach. Secondly we present a unifying method of these algorithms, giving a general application scheme. Finally we give some results, and conclude.

2 CAD Information Recovery

2.1 Object Recognition

Recent developments in object recognition techniques [9], [6], [7] have led to consider that there might be more to the visual cues than just image signal information. Indeed, in [8] we develop methods allowing to recover CAD data from pre-treated image scenes. The results shown in this paper were of a general order, and the presented methods had the inconvenience to be mutually complementary, solving the problem only in areas where appropriate hypotheses on the scenes could be assumed. We now present a unifying approach, integrating both methods into an operational recognition scheme. We shall first briefly recall the approaches described in [8].

Considering the two recognition methods described in [9] and [6] it is possible to recover CAD data from pre-treated images. Both methods are similar in the sense that both match local descriptors between images for recognition using an indexing technique. The descriptors are chosen for their invariance (or quasi-invariance) to viewpoint changes. Once matched between two images, a global verification using a Hough-like vote, allow the filtering of outliers, mismatches and noise. So as a summary of both methods :

1. An unknown image, containing one or several of the model objects, is presented to the system. The objects in this image need not necessarily be in the same position as the model objects. A certain tolerance in viewpoint change is allowed. Similar image cues are extracted, and compared to those in the associative memory. The result of this comparison - complexity $\mathcal{O}(q)$, where q is the number of image cues - is a list of possible matches between several of the model images and the presented, unknown image.

2. A global criterion is used to filter out incoherent matches. According to the method, this varies in application :

   (a) Schmid uses a robust local neighborhood verification, requiring that 30% of the cues of the image in a given zone are matched to model cues in a similar zone.

   (b) Lamiroy implements a generalized Hough transform to select the best similarity transform approximation for the apparent motion between the models and the image.

   This response - i.e. highest number of coherent matches - is used as a classification distance between the models and the image; the highest response corresponding to the most similar model.

[9] primally uses signal based information, the interest point extraction being an enhanced Harris detector. Therefore the method is sensible to image intensity, illumination and light source orientation. The advantage of the method is mainly that, due to the fact that preliminary extraction of interest points is robust, and furthermore the fact that the local descriptors are extremely selective, the approach is very precise and robust in the area of its application. This area can be defined as the one using images with a constant illumination source, textured objects with no varying texture. ‘ [6] has a radically opposed approach where the local descriptors are concerned. Based on an

initial line segmentation of the image, this approach suffers from the major drawbacks image segmentation presents : medium precision, problems of over- and under-segmentation, as well as different kinds of particular problems where occlusion is concerned. Its advantages consist in the fact that segments are a more abstract kind of information with a higher semantic content. Furthermore, the recognition method does not rely directly on image texture and/or intensity (although the preliminary segmentation process does), and is therefore less influenced by a variation of these. Its domain of application tends more towards recognition of objects in a clean and structured environment with clear edges.

The main force of both methods, however, mainly resides in the fact that both solve for recognition (or identification) and matching of the image cues at the same time. Since both approaches basically have the same way of solving the problem, the image cues set apart, one can find a simple way to summarize both methods into a single one, merging both domains of application. Suffices to introduce a geometric constraint into the Schmid approach [9] in order to get a mutually coherent voting system.

Algorithm :

- Extract image cues : since both methods rely on different image cues, we extract both types : interest points for the Schmid algorithm [9], line junctions for Lamiroy [6].
- Match image cues : the image cues are matched using the standard indexing technique described above. The result is a list of couples of image cues that form plausible matches.

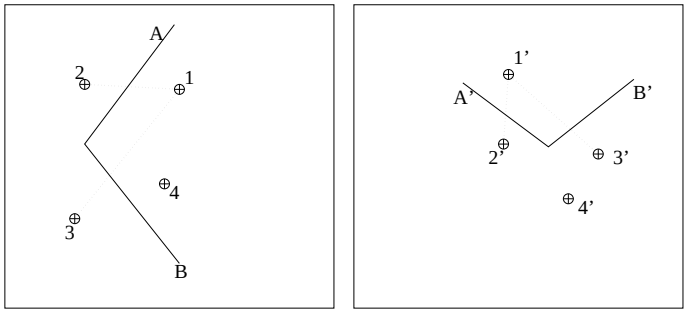

Figure 1: Example of compatible and incompatible matches, both segments matches $A-A'$ and $B-B'$ as well as the connected interest points induce the same motion, while the mismatch $4-4'$, combined with any other point match will induce an incoherent motion.

A supplementary step is now added to the [9] algorithm : instead of checking explicitly the neighborhood of each point for coherent matches, the entire neighborhood is used to form matched couples. Each of these couples induces a similarity movement between the image and the model it matched. This movement being identical to the one used in the [6] algorithm, it can be used to express a vote in the corresponding Hough space. In other terms, given the configurations expressing individual matches between a particular model and the image in FIG. 1, each match between a configuration of segments ($A-A'$ and $B-B'$) define a unique similarity transform. Similarly, for a given match of interest points (e.g. $1-1'$) each couple of matches ($1-1', x-x'$), where x belongs to the near vicinity of 1, defines a unique similarity as well. It is clear that coherent couples, like ($1-1', 2-2'$) and ($1-1', 3-3'$) will define similar transforms, while mismatches like ($1-1', 4-4'$) define transforms that are quite different.

- The voting stage, now consists of two inputs, one coming from the Schmid algorithm [9], the other from the Lamiroy [6] algorithm. Since coherent votes will tend to form clusters in the similarity transform parameter space (or Hough space), voting will consist in enumerating the transforms defined by all matches in the parameter space of the appropriate model.
- The result of the voting process is an ordered list of models, the ones having received the most coherent votes having the more dense clusters in their Hough space, and thus corresponding to the more likely models.

3 Recovering the CAD Data

Given a image base of pre-treated model images we shall call B, and given an unknown image called I, we need to answer the following questions : *Q1="What scene (or model image of B) I belongs to ?"*, *Q2="What CAD objects can be found in this scene, and where do they project in I ?"*. We answered to *Q1* in the previous paragraph. *Q2* corresponds to the recovery of the CAD data that was embedded in the model images.

We propose a geometric approach to the recovery of the CAD data. Once we've matched a model image to the unknown image, we should be able to calculate a relationship between the two images. By expressing this link between them with a geometric relation,

we only need to apply this geometric relation to the CAD information in the model image to retrieve its location in the unknown image. There are two ways of recovering this kind of relationship. One consists in using the matches between the image and the found model during the voting process of the previous phase to estimate an homography between the two images, the other uses the matches between the image cues of the model and the image to estimate the epipolar geometry between the unknown image and its two closest models, using the well known trilinear relation between the three to predict the position of the CAD data in the former.

The similarity obtained from the voting stage is an approximation of the real transform between the model and the image, as it is known that in the general case there exists no geometric relation between two images of a same scene. An approximation by homography, however, can reveal itself of a sufficient nature if the following conditions are met :

- The general scene is sufficiently "flat" in order to have a global transform that is relatively close to the approximated homography. This is because, for two images of a plane, there does exist a projective relationship between the two, called homography.

- The CAD data represents objects that are either sufficiently "flat", or that are small enough compared to the whole scene, not to be affected by the projective aberrations one can expect.

These conditions are general enough to be applicable in a large number of applications. Generally, in the case of automatic assembly/disassembly, the camera can be freely positioned if conserve weak perspective, and if the observed scene is relatively simple, "flatness" can be assumed.

One could argue that the approximation by homography only holds for a limited number of situations, and that, as an attempt to model the apparent motion between images, better models exist. This is true. However, since there exists no exact relationship between two images, another constraint needs to be introduced. The trilinear constraint uses a third image to obtain an exact result for the transform between two images and a third one. In our case, this would mean that we match the unknown image to its two closest models to obtain the required three images. However, one must bear in mind that the process involved in calculating the trilinear relationship between them is numerically far more instable.

- Firstly, the particular system is quite influenced by unprecise or erroneous matches. The recognition and matching process being of a robust nature, it easily tolerates a certain number of mismatches between the image and the models. This pinpoints a fundamental difference between the two approaches. Furthermore, the recognition and matching system relies on few but informationnaly rich points, while the estimation process for the trilinear relations preferably requires a high number of points increasing the gap between the two.

- Secondly, one needs two model images to be matched to the unknown one. This can be considered as a constraint when the number of model images available is low, or when different model images are sufficiently similar to introduce an ambiguity into the recognition and matching process. Moreover, the fact that two model images are involved increases the risk of having mismatches, and causing the estimation process to fail.

We shall show results of both approaches in different situations at the end of this paper.

4 Visually Controlling a Task

Now we've obtained the positioning of the CAD object of interest in our current image we can use this information to guide a robot to execute a specific task. Calculating the 3D position of the object [3, 4] or the use of visual control systems such as [5] do not fall directly in the scope of this paper, but the interested reader can refer to the cited publications. The use of our approach is quite directly adaptable to the cited systems, most of which assume a manual identification of the visual goal. The use of our approach could render these applications semi-automated, the manual definition of the visual goals being restricted to the model images only.

5 Some Results

In this series of tests, we took about ten images of an air filter of a car engine. We then took another image, and compared it to our model base as described above in order to retrieve the closest model image. The result of this recognition step is shown in Fig 2.

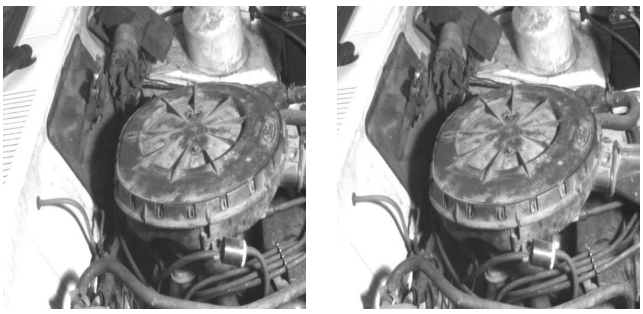

Figure 2: Unknown image (left) and model image found to match the unknown image (right)

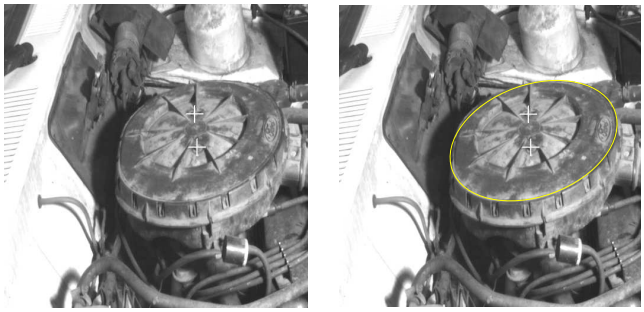

Figure 3: Localized screws in the unknown image (left), and the model image with superposed CAD Data, delimiting the air filter and the screws.

Furthermore we added visual CAD data to our model images (as shown in FIG 3).

We propose to localize two screws on the top of the air filter, and this with the two methods described above : first using the estimation of the best approximating homography, secondly by using the trilinear constraint between the two closest model images and the unknown image. For the first case, we recall that the real motion between the images is not an homography, but that the difference in viewpoint is small enough to assume that this is a decent approximation.

1. We did not represent the results of this method here. We obtained the location of the CAD objects (screws on top of the filter) within 5 pixels in the unknown image.

2. We retrieved the point matches between the unknown image and its two closest models and calculated the trilinear relations between them. As shown in FIG. 3, we obtain a localization within 1 pixel of the correct position, thus greatly improving the previous results.

6 Conclusion

We presented a method allowing to identify CAD data in real images taken from an unknown viewpoint. It requires the identification of this CAD data in some key images that model the span of viewpoints the unknown image may cover. We showed that our method allows a fair precision within 1 pixel. This method is particularly suited for automatic or computer assisted disassembly tasks. Indeed, with our method the system only needs to acquire a number of key views allowing it to recognize the objects it is to manipulate. It is then possible to recognize objects in any position, and to precisely locate parts that need to be operated on. From there on, known algorithms can predict the 3D position and orientation of the parts and guide a robot to undertake the necessary actions.

References

[1] C.H. Chen and P.G. Mulgaonkar. CAD-based feature-utility measures for automatic vision programming. In *Direction in Automated CAD-Based Vision*, pages 106–114. IEEE Computer Society Press, 1991.

[2] D.J. Clemens and D.W. Jacobs. Model-group indexing for recognition. In *Proceedings of* DARPA *Image Understanding Workshop, Pittsburgh, Pennsylvania, USA*, pages 604–613, September 1990.

[3] D. Dementhon and L.S. Davis. Model-based object pose in 25 lines of code. *International Journal of Computer Vision*, 15(1/2):123–141, 1995.

[4] F. Dornaika. *Contributions à l'intégration vision/robotique : calibrage, localisation et asservissement.* Thèse de doctorat, Institut National Polytechnique de Grenoble, LIFIA–IMAG–INRIA Rhône-Alpes, September 1995.

[5] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. IEEE *Transactions on Robotics and Automation*, 8(3):313–326, 1992.

[6] B. Lamiroy and P. Gros. Rapid object indexing and recognition using enhanced geometric hash-

ing. In *Proceedings of the 4th European Conference on Computer Vision, Cambridge, England*, volume 1, pages 59–70, April 1996. Postscript version available by `ftp`[1].

[7] B. Schiele and J.L. Crowley. Object recognition using multidimensional receptive field histograms. In *Proceedings of the 4th European Conference on Computer Vision, Cambridge, England*, pages 610–619, 1996.

[8] C. Schmid, Ph. Bobet, B. Lamiroy, and R. Mohr. An image oriented CAD approach. In *Proceedings of the ECCV workshop on Object Representation*, 1996. Postscript version available by `ftp`[2].

[9] C. Schmid and R. Mohr. Combining greyvalue invariants with local constraints for object recognition. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, San Francisco, California, USA*, June 1996. [3]

[10] L. Shapiro and K. Bowyer, editors. IEEE *Workshop on Directions in Automated CAD-Based Vision*, Maui, Hawai, 1991. IEEE Computer Society Press.

[11] M. Tonko, K. Schäfer, and H.H. Nagel. A multi-agent architecture for distributed vision-based disassembly. In *Annual IAR Conference, November 21-22, 1996, University Karlsruhe, Germany.*, November 1996.

[1] ftp://ftp.imag.fr/pub/MOVI/publications/Lamiroy_eccv96.ps.gz
[2] ftp://ftp.imag.fr/pub/MOVI/publications/Schmid_WSeccv96.ps.gz
[3] ftp://ftp.imag.fr/pub/MOVI/publications/Schmid_cvpr96.ps.gz.