An experimental measurement system for industrial inspection of 3D parts

Claus Brenner, Jan Böhm, Jens Gühring[a]

Institute for Photogrammetry (ifp)
University of Stuttgart
Geschwister-Scholl-Straße 24, D-70174 Stuttgart, Germany

ABSTRACT

A research group at the University of Stuttgart has set up an experimental measurement robot for industrial close range inspection. During a test run, the feasibility of a multi-sensor/actor system has been shown. The system uses optical sensors to perform different tasks including object recognition, localization and gauging. It is a step towards systems which are able to inspect and gauge several parts from a set of parts stored in a 3D model database. This paper describes the results which have been obtained so far and were demonstrated during a test run. It then focuses on our latest developments concerning 3D data acquisition, registration, segmentation, model generation from CAD data and object recognition.

Keywords: industrial inspection, range images, object recognition, CAD

1. INTRODUCTION

In the last 20 years we have seen dramatic changes in production methods for industrial goods. The advent of general purpose industrial robots made it feasible to have the same robot for different tasks or across different products. It became evident that, wherever practical, it is more economical to have non-specialized production units. This way, the cost of change can be kept small after a product has been redesigned. Nowadays, the situation is characterized by two contrary developments: part complexity increases and production lot size decreases. Time-to-market is more important than ever before. The decrease in development cycle time is obvious in many consumer product areas including automobiles and electronic devices.

Apart from changes in production, all other steps of the product cycle are affected as well. Considering product design, parts are modeled in 3D using feature based and parametric CAD systems, which allow rapid changes. Part design is evaluated at early stages for technical soundness by simulation methods. Aesthetic quality is judged by the early fabrication of models that look almost like the final product, which has become feasible using rapid prototyping techniques such as stereolithography and vacuum moulding.

However, looking at quality assurance, we find that changes have not been as dramatic as in other areas. Still, in many cases part geometry is checked against the specification by individually prepared gauges or specialized measuring systems. Sometimes, random samples are drawn and measured by coordinate measurement machines (CMM). Facing the trend towards a 100% quality control, it is obvious that those techniques are too expensive and provide not enough flexibility.

Optical measurement techniques, on the other hand, have several properties which make them ideally suited for flexible gauging and inspection tasks: they are able to measure thousands of points in a matter of seconds; they are applicable to a wide range of materials, including deformable objects; and they can yield very accurate results when used in conjunction with proper calibration techniques. Moreover, optical techniques can capture other important object features like transparency, color and surface gloss. And, since the object is captured anyways, some simple vision tasks like reading a barcode on the object or checking for completeness can be done without the expense of additional sensors.

Despite their advantages, optical measurement techniques are not very well accepted in industry.[1] One reason for this is that traditional measurement techniques, like CMM's, are well established whereas optical systems with comparable performance have not been commercially available until recently. Also, properties like surface roughness

[a]E-mail: Claus.Brenner@ifp.uni-stuttgart.de, Jan.Boehm@ifp.uni-stuttgart.de, Jens.Guehring@ifp.uni-stuttgart.de

today are defined in terms of CMM measurement results, which necessitates the definition of an "optical equivalent" before market acceptance can be expected. Another drawback is that optical techniques are considered to be too complicated to operate under factory conditions. Furthermore, optical measurements often give accuracies which depend on the specific object. In unfavorable cases, for example if an object's surface is soiled, measurement may become impossible using fixed sensor and lighting positions. But changing these conditions (e.g. by changing the sensor, lighting or object positions) usually requires some skilled person familiar with that particular measuring system. Therefore, the measurement systems that have made their way into industrial applications usually use very stable features such as retroreflective targets[2] or very well controlled environments.[3]

However, in our opinion, the slow industry acceptance of optical 3D measurement techniques is not a vote against those techniques but rather reflects the standard learning process in industry. Unfortunately, heavy competition and outsourcing of product development to supplier companies often limit the research horizon to one or two years. On the other hand, we see a parallel to vision systems used in 2D inspection such as number recognition and completeness tests. Due to technical progress in the fields of camera (CCD and CMOS high dynamic range technology, "intelligent" cameras with integrated processing) and software technology (standardized image processing modules, reliable implementations), nowadays vision applications can be developed fast (i.e. cheap). Consequently, we can see a growing number of manufacturers and installations for 2D vision systems.

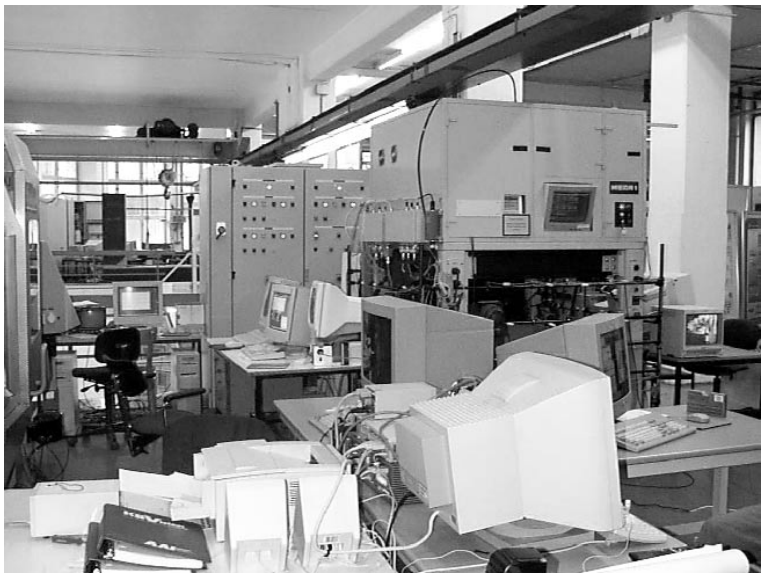
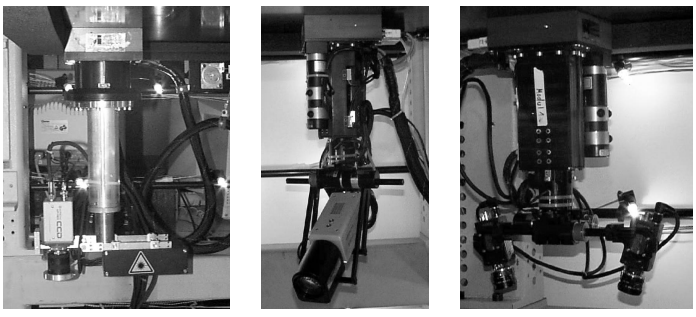


Figure 1. View of the experimental measuring platform during a test run. Foreground: several general purpose computers used for sensor specific processing, image processing, system and machine control. Background: control cabinet (left) and measuring platform (right).



(a) (b) (c)

Figure 2. (a) Laser projector. (b) Multi-parametric camera. (c) Stereo camera.

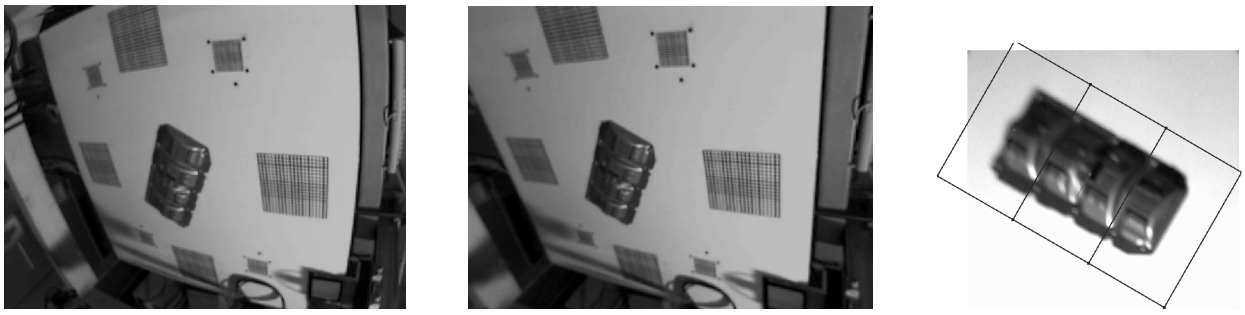
To develop techniques in the area of 3D inspection and measurement, a research group has been funded by the German Research Foundation for a period of three years. Five institutes of the University of Stuttgart were involved, including mechanical and optical engineering, photogrammetry and computer science. As a result of this work, research goals have now been extended and a special research field was initiated in January, 1998. Now, seven institutes will continue this long term research for an expected duration of nine years.

2. EXPERIMENTAL CONCEPT VALIDATION

In order to validate the concepts on 3D inspection developed by the research group, an experimental measuring system has been set up. It will be adapted continuously to the needs of the group. However, during a test run, the group was already able to demonstrate that handling of such a complex system is feasible. The demonstration included the tasks of calibration, object identification and localization, and measurement.

The experimental measurement system currently consists of the following sensors (Figs. 1, 2):

1. a laser projector which is used to obtain height maps via the coded light approach
2. a multi-parametric 3-chip CCD camera which has the ability to change parameters like focus, focal length, aperture and several electronic parameters, based on the interpretation of the image
3. a stereo camera which employs two standard CCD cameras
4. a wide angle CCD camera used for capturing the entire measurement volume.



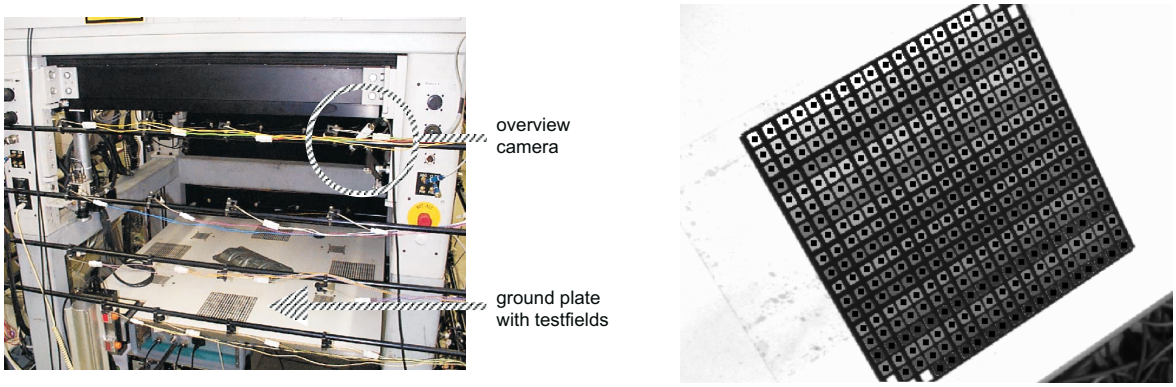
(a) (b) (c)

Figure 3. (a) Original image taken with the overview camera. (b) The same image after correction of lens distortion. (c) Again, the same image after rectification. Each image pixel now corresponds directly to a certain metric area on the ground plate. The object is segmented and three locations for the stereo camera are generated (boxes correspond approximately to the field of view of one stereo camera image take).

The size of the measuring volume is about $1000 \times 1000 \times 700$ mm^3. Lighting is provided by four light source arrays with 16 individually controllable lights each. All sensors (except the overview camera) were mounted on actor modules with 3 (laser projector) or 5 axes (multi-parametric camera, stereo camera), totaling 13 axes. The actor modules are held magnetically at the ceiling of the measurement volume. An air cushion is formed between the ceiling and the modules; they are horizontally moved according to the Sawyer motor principle where the ceiling forms the stator. This linear stepping motor design allows for a very precise positioning. All axes are controlled by a VME-bus system which in turn is connected to a PC that receives positioning requests via the parallel virtual machine (PVM) protocol. The entire setup consisted of about 10 computers of different platforms (PC, Silicon Graphics, Sun). Data and control information is transferred using a standard ethernet network and the PVM communication protocol.

The demonstration incorporated the following steps which were performed by the system automatically:

1. The coarse location of the object using an overview camera. With a focal length of about 4 mm, this camera was able to capture almost the entire measurement volume. Images were first corrected for lens distortion using digital image processing and lens distortion parameters taken from a pre-calibration step (Fig. 3(a–b)). Then, they were rectified (using four point correspondences determined after system setup). Using image processing,



(a) (b)

Figure 4. (a) View of the measuring volume. The overview camera is in the top right corner. On the ground plate, four small and four large testfields can be seen. Between them, the metal sheet object to be measured. (b) Result of automatic testfield evaluation. This image shows the rectified image overlaid by the detected target areas and measured target centroids.

the object was segmented from the background formed by the white ground plate of the measurement volume. Based on this segmentation and the known field of view, three positions and orientations for the stereo camera were generated (Fig. 3(c)).

2. Calibration of the ground plate coordinate system. In order to position the stereo camera, the transformation between the machine coordinate system and the ground plate coordinate system has to be established. For this purpose, the ground plate contained several testfields. These fields consist of a specially striped pattern which allows the fully automatic determination of point numbers and subpixel accurate point location of a large number of points (Fig. 4). The reference positions of all testfield points were determined in advance by bundle calibration. During the demonstration run, the camera moved over four testfields (given by approximate machine coordinates) and the resulting images were evaluated. Then, the transformation between the machine coordinate system and the ground plate system was computed.

3. (More precise) object location. Using the stereo camera, images were taken based on the positions and orientations generated from the coarse location step. Those images were merged by image processing to obtain a single synthetic image (Fig. 5). Again, image segmentation was used to separate the object from the background (Fig. 6(a)).

4. Object recognition. This step was done using a hypothesize-and-test approach. Based on the scene segmentation (Fig. 6(a)) and an object model which contained a polygonal description of the object's edges, hypotheses were generated for the possible object location and orientation (Fig. 6(b)). All hypotheses were refined using a iterative closest point algorithm similar to the method described by Besl and McKay.[4] In our case, however, a projective transform was estimated rather than a 3D transformation. After refinement, the solutions were compared against the image segmentation and the best solution was selected (Fig. 6(c)). The transformation defined by this solution was used to render a synthetic 3D view of the scene consisting of the calibrated target positions (which define the world coordinate system) and the object model transformed into the world coordinate system (Fig. 6(d)).

5. Measurement tasks. After the object is located, dedicated measurements took place at measurement locations which were defined in advance relative to the object model. One of them was the measurement of a drilling hole using the light array to minimize shadows. The other was to acquire a dense height model of a certain part of the object using the coded light sensor. Those tasks will not be described in more detail here.

3. OBJECT RECOGNITION USING 3D CAD MODELS

The demonstration described above was an important step for us, since it showed that a complex measurement platform can be realized – albeit it was no "industrial strength" setup. Also, basic concepts were realized such as

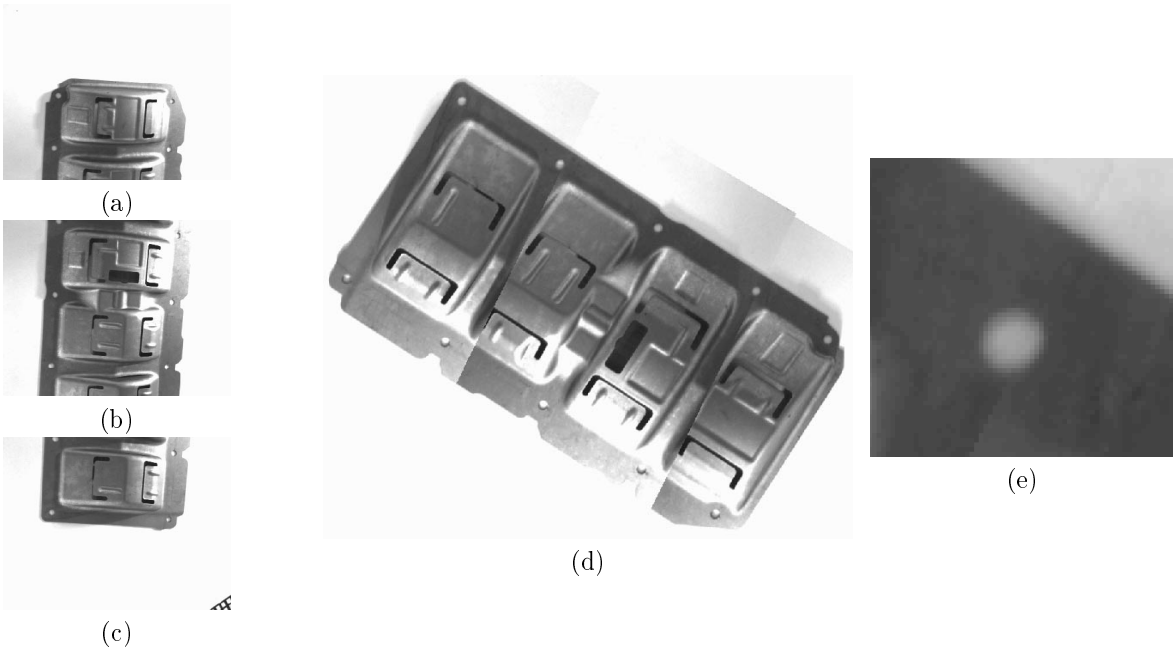


Figure 5. (a)–(c) Single images taken by the stereo camera. (d) Synthetic image obtained by merging images (a)–(c) using the known exterior orientations. (e) Detailed view of a seam in the synthetic image where two single images meet. No geometric discontinuities can be discerned.

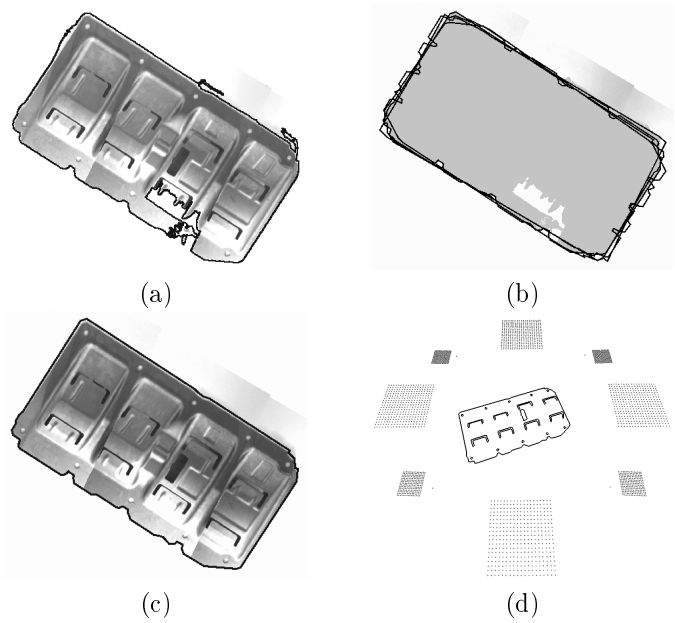


Figure 6. (a) Segmentation of the synthetic image. Note how the segmentation is erroneous due to bright (glossy) areas on the metal surface. (b) Four hypotheses generated for the possible object location and orientation. (c) The best hypothesis is selected (which is the correct solution). (d) Synthetic 3D view of the object model transformed into the scene.

a coarse-to-fine strategy (overview to detailed image) and an object recognition based on a hypothesize-and-test approach. However, many of the solutions that were used rely on 2D information rather than on 3D information. For example, the overview image is rectified which might not work reliably for objects with larger heights. As another example, object recognition was based on the polygonal description of the object's outline, which, of course, cannot

be regarded as a general solution. Another problem is that since no CAD model was available for the complex metal sheet object used for our experiments, the object model was constructed by photogrammetric techniques. Ideally, it should be derived automatically from the CAD model.

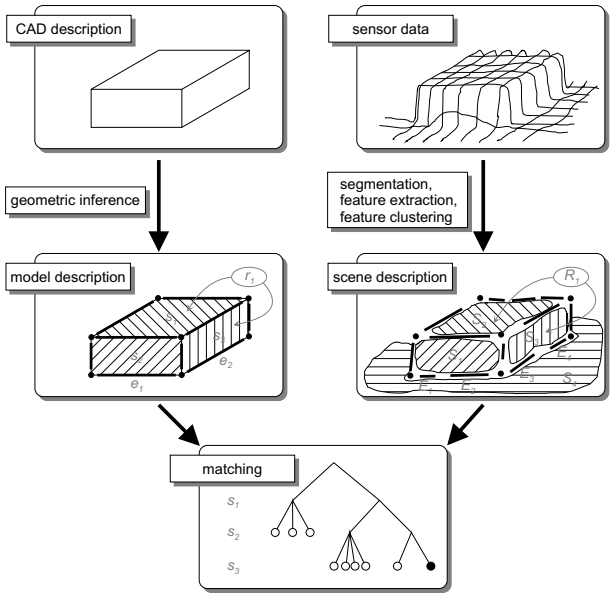


Figure 7. Typical object recognition system. The left part of the diagram can be solved in advance (off-line) whereas the right part has to be solved each time a new object is to be recognized (on-line).

The typical approach of object recognition systems is depicted in figure 7.[5,6] From this, we can determine the following tasks which have to be addressed:

1. The definition of object models which are useful for object recognition
2. Scene segmentation and feature extraction
3. Matching the model and the scene.

3.1. Deriving Models from 3D CAD Data

The importance of the modeling step has often been underrated. In the context of industrial parts, one can use specific models, so the problems associated with parametric or generic models can be avoided. Still, there are many possible choices for the features and attributes which can be used.

Using global object properties such as volume, roundness or higher order moments, objects can be described by a single parameter vector. Matching objects to models then is reduced to a comparison of parameter vectors using some measure for similarity. Although these global approaches are used frequently for simple vision tasks, they are generally not considered to be robust enough, particularly in the presence of occlusion and clutter.[7]

Thus, the approach used by many researchers uses geometric primitives, like points, contours, surfaces and volumes. Matching is accomplished by establishing a number of feature correspondences between an object model and the scene. Non-global features and feature properties ensure that a match is still possible even when parts of the object are occluded or are not present in the scene at all.

The selection of useful features is another important topic. Since the complexity level of features can be chosen, one is free to adopt either sensor-specific or model-specific, low-level or high-level features. Sensor specific and low-level features might produce features that are easily detectable in the scene; however it is their low information content and abundance which makes subsequent matching steps computationally expensive. Model specific and high-level features can be derived easily from the model and their rareness guarantees a low complexity of the matching algorithm; on the other hand, their extraction from the scene might be hard or impossible. For example, it is so far

not possible to use a CAD representation directly for matching, because automatic extraction of the high-level CAD objects from sensor data is not feasible.

Since it is nevertheless desirable to obtain object models automatically and a growing number of industrial parts today is manufactured using CAD/CAM technology, several researchers have used CAD data as a basis for object models.[8,9,6] This way, CAD data is not used *directly* but rather an intermediate processing step is used to "enrich" CAD data and derive explicit descriptions for what is given implicitly in the CAD data.

In our approach, we chose a surface description based on planar faces as the common domain for model and scene representation. To obtain this description from the CAD model, we used a distinct feature of the CAD system Pro/Engineer: From the CAD system, triangulated surfaces can be exported with triangles grouped into faces. This way the segmentation information already present in the CAD system is maintained. We then compute features based on the triangulation. This set of features forms our model.

3.2. Segmentation and Feature Extraction

The sensor data from different views is first integrated into a single triangulated object representation. It is then segmented into planar regions by a region growing algorithm which operates on triangulated surfaces. The neighborhood of the triangles is used to determine the direction of growing. After each iteration of the algorithm the plane parameters are estimated using principal components analysis. The algorithm utilizes the triangle's normal vector and the distance of the triangle vertices to the estimated plane to determine the homogeneity of the planar patch.

For each face the following set of features is computed: plane equation, normal vector, 3D boundary polygon, 3D center of canonical bounding rectangle, transformation into 2D, 2D boundary polygon, 2D bounding rectangle, area, circumference, maximum distance in region, and bounding box width and height.

3.3. Object Recognition by Feature Matching

We will now address the task of matching the model and the scene. Let us suppose we are given a set of features $S = f_1, f_2, ..., f_n$ describing the scene and a second set $M = F_1, F_2, ...F_N$ describing the model. The task of the object recognition step is to find a global correspondence, i.e. a set of scene-model pairings $(f_1, F_{m_1}), (f_2, F_{m_2}), \ldots$ where each feature in the scene is grouped with its corresponding feature from the model. Theoretically, there are $\mathbf{O}(N^n)$ different possibilities of such pairings. Of course, only a fraction of those constitute valid solutions. In order to limit combinatorial explosion, researchers have used different techniques like *generalized Hough transform*, *graph matching*, *maximal cliques*, *relaxation labeling* and *constrained tree search*.[10]

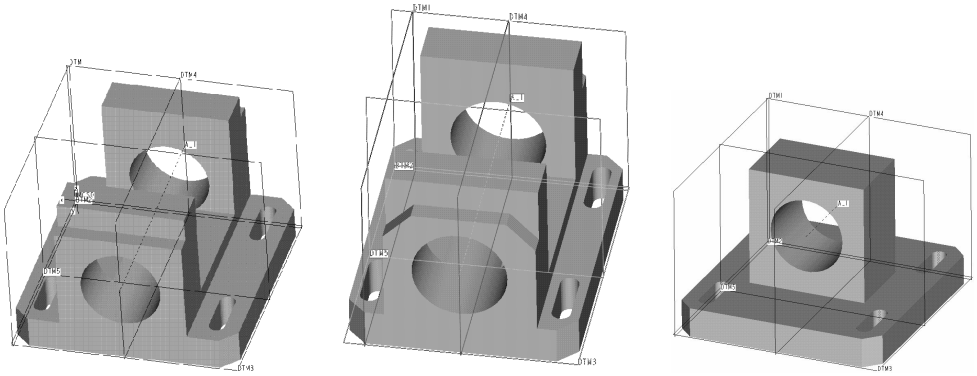
To verify the correctness of a *global correspondence* we can compute one rigid body transformation from all pairings and check whether each single scene feature is correctly transformed onto its corresponding model feature. To check the consistency of a *local correspondence* we can apply so called geometric constraints. Unary constraints are fulfilled if a scene-model pairing is locally consistent. Examples are line length, circle radius, curvature, circumference, bounding box dimensions, area and volume. Binary constraints check whether a pairing is mutually consistent with all other pairings established at that point. The angle and the distance between two patches of the scene compared to the same relations of the corresponding features in the model can be used as binary constraints.[7]

In our first experiments, we used *graph matching*, which is one popular way used in conjunction with region oriented segmentation. The result of the segmentation step is stored in a region adjacency graph (RAG), where each vertex represents one region. Two vertices are connected by an arc if the corresponding regions are adjacent. The model information is stored in the same manner. We now search for a projection of the vertices of the one RAG onto the other which preserves the topology of the graph, i.e. two vertices (not) connected by an arc in the one RAG should (not) be connected in the other. In other words, we are looking for a subgraph isomorphism. Of course, since we are dealing with sensor data we can not expect to find an ideal match of all vertices, therefore we have to allow for some error correction such as the addition and removal of vertices and edges. Vertices as well as edges can carry attributes to further reduce the search space by applying geometric constraints. However, since only adjacent regions are connected by arcs only they will be considered for the binary constraints. Therefore not all of the geometrical information available is used. One problem of graph matching is the computational cost. Since the problem is NP-complete, the time generally grows exponentially with the number of vertices and edges. One way to reduce this cost is to partition the set of vertices into distinct subsets, also known as labeling. This can drastically reduce the search space and thereby save time. We have used graph matching in previous studies,[11] but have found it difficult to cope with computational cost. Because it is hard to find any sensible labeling of the vertices if all

features are of the same kind (e.g. planes) we sometimes experienced worst case behavior. We were only able to match very small subgraphs (less than 10 vertices). Furthermore, adjacency is not a very stable characteristic in 3D segmentation. Problems in the alignment process and shortcomings in the segmentation both contribute to this.

Because of this experience, we decided to follow the *constrained tree search* approach in this work. Considering the notation as introduced above, we first start with one scene-model feature pair (f_i, F_{m_i}). For each possible pairing we find, search goes on to the next level, where a correspondence for the second feature is sought. This leads to the matching path $(f_i, F_{m_i}), (f_j, F_{m_j})$. This search proceeds in a recursive manner. It is well-known as depth-first search. Again, the computational cost grows exponentially with the number of features. The key for finding a solution in a reasonable amount of time is the choice of the constraints used to bound the branching in the tree. For each scene feature only a subset of model features is selected as possible matches using unary constraints. At each stage of the search process we check the binary constraints of the current pairing with all pairings along the path to the root. In our experiments, we found that the circumference and the area of a surface patch are not ideal candidates for determining local consistency. The circumference of a segmented patch in sensor data tends to be much larger than that of an CAD model. In contrast, due to holes in the sensor data and segmentation, the area tends to be much smaller. The attributes (width and height) of the canonical bounding box and the maximum distance within a region provide excellent means to reduce the number of possible pairings. In the experiments conducted so far, we were not confronted with cluttered scenes. What we do have are slight deviations of the manufactured part from the CAD model. Therefore, we can not expect to find a correspondence for all model features and vice versa. This problem is increased by the difficulties in the segmentation stage where a great number of small "sliver regions" is produced which do not correspond to any model feature. To deal with this situation in the context of computation time, we first sort the scene features by size (area) and then prune the set of features to the size of the set of model features. During the search, we allow the skipping of features, i.e. if no correspondence can be found for a certain scene feature we remove this feature for the current matching path and continue with the next.

4. EXPERIMENTAL RESULTS



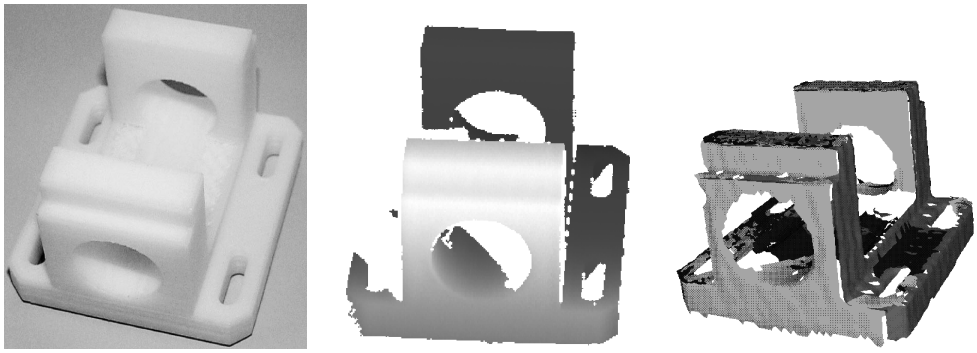
(a) (b) (c)

Figure 8. (a) Object "bearing 1" as constructed using the CAD system Pro/ENGINEER. (b) Object "bearing 2" constructed with minor changes. (c) Object "bearing 3" with major changes.

For first experiments, we decided to construct and build an object ourselves. We used the CAD system Pro/ENGINEER to construct a fictitious object "bearing 1" with dimensions $75 \times 70 \times 50$ mm^3 (Fig. 8(a)). The object was then manufactured directly from CAD data using a rapid prototyping machine from Stratasys, Inc. This machine builds the object layer by layer, applying in each layer a thin stream of ABS material pressed through a nozzle (Fig. 9(a)). This technique is called fused deposition modeling; accuracies down to 0.1 mm can be obtained.

Using an ABW structured light projector with 320 lines,[12] 13 range images of the manufactured part were taken (Fig. 9(b)). The images were aligned (Fig. 9(c)) and merged using the POLYWORKS package from InnovMetric Software, Inc.[13,14]

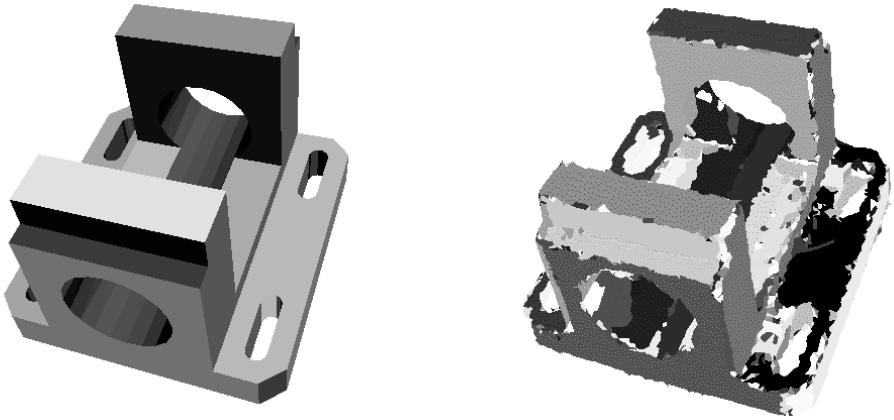
The CAD model and sensory data are processed in the way described above. We compute a segmented CAD model and a segmented scene model and feature sets for both. Segmentation time is less than a minute for 34,000



(a) (b) (c)

Figure 9. (a) View of the object "bearing 1", manufactured by a rapid prototyping machine. (b) Example of a range (depth) image obtained with an ABW structured light projector. The missing part of the object on the left hand side is typical for triangulating sensors and results from shadowing. (c) 3D view of the alignment of three range images (light gray, dark gray, black).

triangles on a 400 MHz Pentium II PC. Figure 11 shows examples for corresponding faces in the model and the scene and table 1 lists some of the attribute values.



(a) (b)

Figure 10. (a) Segmented surface description obtained from the CAD model. (b) Result of segmentation of the merged sensor data into planar faces.

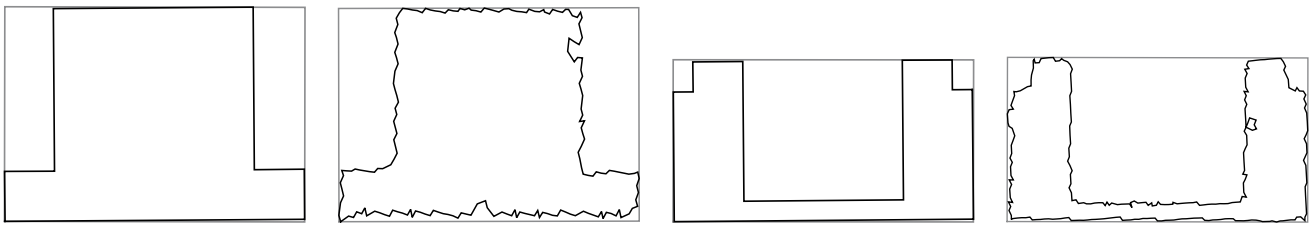


Figure 11. Example for two faces as they appear in the model and as segmentation result obtained from the scene.

feature	m-0	s-0	m-1	s-1
area	1417	1072	1525	1261
maxDist	66	64	82	78
bBox-w	60	60	75	73
bBox-h	43	43	41	40
circumference	205	244	300	327

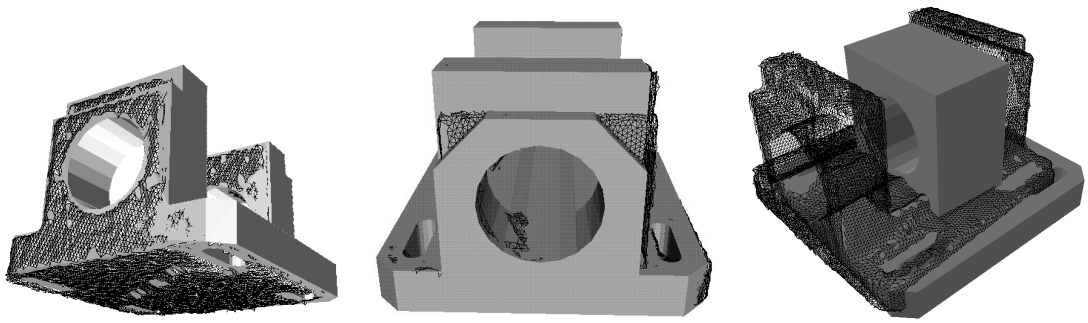
Table 1. Numerical values for some attributes of the two faces shown in figure 11. Abbreviations: m=model, s=scene, maxDist=maximum distance in region, bBox-w, bBox-h=bounding box width and height.

Matching is done using constrained tree search. After several faces are matched, the "best" (in least square sense) rigid body transformation between scene and model is estimated using a non-iterative approach described by Sanso.[15] As a measure for the quality of the match, both the standard deviation of the parameter estimation and the deviation of scene points from the object's surface can be used (Table 2). Figure 12 shows the superposition of scene and model when the transformation computed by the matching is applied.

We also conducted experiments with different CAD models. One model is a slight modification of the original object "bearing 1", while the other is drastically different ("bearing 2" and "bearing 3", see Fig. 8). The matching also yields good results in terms of RMS error for the modified model, since only the RMS error of corresponding features is computed. For "bearing 3" we were only able to get a match when we set matching thresholds to unreasonably high values. However, RMS values for this case differ clearly from the other two cases (Table 2).

match	RMS-1	RMS-2
scene-bearing 1	1.112	1.323
scene-bearing 2	1.112	1.396
scene-bearing 3	5.139	100.466

Table 2. Assessment of matching quality. RMS-1 is the RMS difference of the features that were used for matching. RMS-2 is the RMS difference for all scene points.



(a) (b) (c)

Figure 12. Superposition of model and scene, with applied transformation from the matching. (a) for object "bearing 1". (b) for object "bearing 2". (c) for object "bearing 3". The deviation between the object and the CAD model is clearly visible.

5. CONCLUSION

We have reported on the ongoing work on a measurement system for inspection and gauging of industrial parts. We have shown the results on system integration which were demonstrated during a test run. We also reported on recent results regarding model generation from CAD data, segmentation and object recognition.

In the future, we will concentrate on more complex models involving free-form surfaces, the integration of grayvalue and color data, and new matching approaches.

ACKNOWLEDGEMENTS

We wish to thank M. Ebenhoch of the Institute for Machine Components for manufacturing the test part, G. Hetzel of the Institute for Parallel and Distributed High-Performance Systems for making available the ABW coded light projector and all our colleagues from the research group. This research was supported by the German Research Foundation under research grant DFG-Le525/4 and DFG-SFB514.

REFERENCES

1. A. Grün, "Digital close-range photogrammetry — progress through automation," in *Proc. Comm. V Symp. Close Range Techniques and Machine Vision, IAPRS, vol. 30, part 5*, pp. 122–135, 1994.
2. H. A. Beyer, "Digital photogrammetry in industrial applications," in *From Pixels to Sequences, IAPRS vol. 30 part 5W1*, E. P. Baltsavias, ed., pp. 373–378, 1995.
3. W. Bösemann and K. Sinnreich, "An optical 3D tube measurement system for qualitiy control in industry," in *Automated 3D and 2D vision. SPIE Vol. 2249*, pp. 192–199, 1994.
4. P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **14**(2), pp. 239–256, 1992.
5. B. Bhanu and C.-C. Ho, "CAD-based 3D object representation for robot vision," *IEEE Computer* **20**(8), pp. 19–35, 1987.
6. P. J. Flynn and A. K. Jain, "CAD-based computer vision: From CAD models to relational graphs," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **13**(2), pp. 114–132, 1991.
7. W. E. L. Grimson, *Object Recognition by Computer: The Role of Geometric Constraints*, Series in Artificial Intelligence, MIT Press, Cambridge, Massachusetts, 1990.
8. R. C. Bolles and P. Horaud, "3DPO: A three-dimensional part orientation system," *The International Journal of Robotics Research* **5**(3), pp. 3–26, 1986.
9. C. Hansen and T. C. Henderson, "CAGD-based computer vision," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **11**(11), pp. 1181–1193, 1989.
10. O. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, Cambridge, Massachusetts, 1993.
11. J. Böhm, "Objektlageerkennung durch Matching von CAD-Daten mit 3-D Sensordaten," Master's thesis, Universität Stuttgart, Institut für Parallele und Verteilte Höchstleistungsrechner, 1997. Thesis Nr. 1525.
12. T. G. Stahs and F. M. Wahl, "Fast and robust range data acquisition in a low-cost environment," in *Close-Range Photogrammetry meets Machine Vision, SPIE Vol. 1395*, pp. 496–503, 1990.
13. R. Bergevin, M. Soucy, H. Gagnon, and D. Laurendeau, "Towards a general multi-view registration technique," *IEEE Trans. on Pattern Analysis and Machine Intelligence* **18**(5), pp. 540–547, 1996.
14. M. Soucy, "Innovmetric's multiresolution modeling algorithms," in *SIGGRAPH'97 Course Notes, Multiresolution Surface Modeling*, 1997.
15. F. Sanso, "An exact solution to the roto-translation problem," *Photogrammetria* **20**, pp. 203–216, 1973.