Calibration Issues for Projector-based 3D-Scanning

Tilo Gockel, Pedram Azad and Rüdiger Dillmann

Abstract— **In this paper we want to introduce a new approach to 3D scanning of dynamic scenes. This implementation enables the user not only to manually move a scan head over the object to scan but also to capture moving objects. Registration from scan to scan is done in real-time. The user interacts with the system and watches the scene assembling. He can immediately respond on shadings that occur due to undercuts in the scene. The chosen structured light scan method uses a primarily unknown speckle image that can be easily etched on a chrome-on-glass slide and projected using a strobe light. In the current large scale implementation we are using a standard video beamer and a standard digital camera. Miniaturization and adaptation for special purposes (i. e. medical applications) are scheduled for next year. Focus in this paper shall be laid on calibration issues regarding camera and projector**

Index Terms—**Calibration, Pattern Projector, Structured Light, 3D Shape Acquisition, 3D Scanning, Single Shot.**

I. Introduction

3D-SCANNING of large objects using a standard laser- or pattern projector-based scan system can be a very time-consuming process. After each single scan the user has to do an offline registration with the previous scans and has to check for remaining shadows due to undercuts in the scene. A scan system that instantly shows the results during the scan process and therefore enables the user to react interactively would be much more efficient. This type of system can be set up using a pattern projection-based scan mechanism, a fast online registration from scan to scan and a fast visualization of the resulting 3D data.

Perhaps the most famous results of such an approach were published by Rusinkiewicz et al. [5]. These researchers used a multi-pattern projection system with a complex tracking method from image to image. This method requires a video beamer and therefore lacks the possibility of strobe lighting. Furthermore the projector cannot be changed to a simple lamp+slide+optics-system (cp. slide projector). Similar approaches of other researchers using single shot methods based on specific pattern codes depend on a very precisely manufactured pattern slide [3,8-12]. To avoid these drawbacks we want to focus on a single shot / single pattern projection system in our approach with no demands for accuracy of the slide pattern. The test setup uses a standard video beamer projecting *only one* speckle pattern (white noise). This projector can be easily substituted with a simpler device like a low-cost gobo projector known from multimedia applications. A standard digital camera observes the scene.

Normally projector calibration in similar systems is done using a well-known pattern [8]. In our system we had to choose a new approach that enables us to calibrate the pattern projector firstly with only one static pattern and secondly with a pattern that is manufactured without regard to accuracy and is not initially known by the system.

Section II of this paper explains the calibration procedure we have developed and gives a short insight into the methods of pattern recognition (correlation) that were used, and the registration and visualization methods that were adopted. Section III shows the actual implementation, the test-bed hardware and software details. In Section IV results of the system can be found. The paper ends with Conclusion and Future Work in Chapter V.

II. Algorithms and Workflow

A. System Calibration

The calibration procedure of the scan system consists of camera calibration and projector calibration, done in this order so that the calibrated camera can be used to calibrate the projector.

To calibrate the camera we use an extended DLT approach (direct linear transformation - a least squares method) that deals with radial lens distortions [1,4]. The algorithm used calculates the minimization of radial lens distortions using an iterative method, starting with DLT-calculated (distorted) coordinates. This approach is test-field-based and therefore a set of known 3D points in world coordinates is needed. We have realized this using a stack of acrylic plates of known thickness with the top plate printed with dots of known distance. A Region Growing algorithm searches for the dots (ellipses) and calculates the centers.

Projector calibration is done in a similar fashion, but the

Manuscript received November 18, 2003 The authors are with the University of Karlsruhe, Dept. of Computer Science, Institute for Industrial Applications in Informatics and Microsystems (IAIM). Haid-und-Neu-Strasse 12, c/o Technologiefabrik, D-76131 Karlsruhe, Germany. Head: Prof. Dr.-Ing. Rüdiger Dillmann (e-mail: {dillmann, gockel, azad}@ira.uka.de).

calibration pattern is not known. Input for the formerly calibrated camera is a speckle- / white noise pattern as shown in figure 4 which is projected on the base plate. This initial image is used as a template for the remaining parts of the process. The pattern is projected on other plates placed on different heights. A search is performed on the new images to find small local patterns (15x15 camera pixels).

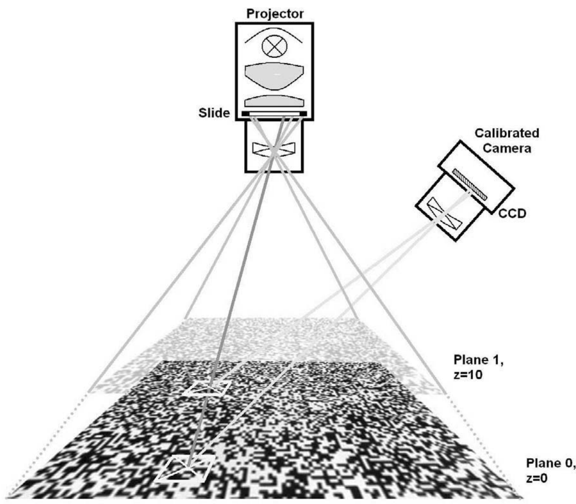

Fig. 1. Schematic sketch of the systems' geometrical layout used for Projector Calibration

The center of each of the local patterns, that are found in this way, forms a point pair together with its corresponding point of the template plane *(z=0)*. These point pairs are used as input for the extended DLT, this time to calculate projector parameters.

Figure 1 shows the intersection of camera rays and projector rays that is used for calculating the position of the small local patterns. To conclude this section on calibration it is worth mentioning that, using this method, a projection of the template onto a plane located between the image plane and template or base plane is calculated. This is not a calculation of a projection of the template onto the image plane. Naturally this makes no difference for the (extended) DLT calibration procedure.

B. Pattern Recognition

Given an image with pattern distortions due to object geometry, a purely correlation based algorithm would not be capable of finding a sufficient amount of correspondences to the template pattern. Implementing the algorithm of Gruen leads to much better results [2,4]. Given two display windows *f(x,y)* and *g(x,y)*, the interrelation between the gray value functions *f(x,y)* and *g(x,y)* is modeled using an affine transformation and a translation. After linearization of *g(x,y)* using a Taylor polynome (1st order) a Gauss least squares approach can be applied. Since *g(x,y)* is naturally non-linear, the calculation of the parameters for shift and for the affine transformation is an iterative process. Figure 2 shows the results of our implementation, which were sufficient for calculating as much 3D information as shown as 3D point cloud in figure 5.

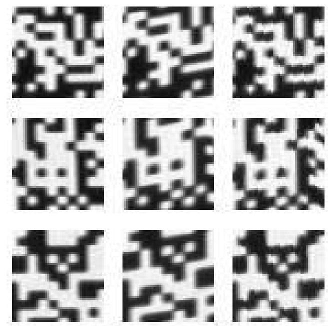

Fig. 2. Three local sample patterns for demonstration of the Gruen Algorithm; each left pattern in a row is a template pattern, each middle one is the actually captured pattern (distorted by scene geometry) and each right one is the pattern after application of the Gruen Algorithm.

Besides the (optional) radiometric parameters of the Gruen algorithm, a local normalization is done as the starting point of the image processing that follows. This is done in order to avoid the influence of an object's texture as much as possible.

C. Fast 3D Registration

One important requirement for the interactive scanning of dynamic scenes is the ability to do a really fast online registration from scan to scan. This shall not be focus of this paper, so we will only give a short insight: the actual implementation uses an icp (iterative closest point) variant from the vtk (visualization toolkit) software library [6,7 and 13].

The trick is to search for good initial guesses using plausibility considerations. Examples of these considerations are 'How fast can the user move object or scan head?' and 'How long ago was the last image taking?' The method is to try the eight basic translations and terminate if no fast convergence is visible. Also attention has to be paid to the closest-neighbour-search part of the algorithm. In the near future we want to change this part of the vtk method towards a kd-trees approach.

D. Online Visualization

A fast online visualization of the actual over-all scan result is very important for this system to provide the interaction that we have described. Visualization is done on point-cloud level at the moment using kitware's visualization toolkit but shall be changed to a triangulated and rendered presentation [13].

Triangulation of single scan shots can be done easily and also very fast for we know the neighbourhood dependencies of the points (resulting in a simple zigzag edge drawing). First tests have shown that vtk is too slow for this purpose and therefore an implementation was done on standard OpenGL. Regarding the visualization approach of Rusinkiewicz [5] using a complex qsplat mechanism for rendering on point

Proceedings of the Shape Modeling International 2004 (SMI'04)
0-7695-2075-8/04 $20.00 © 2004 IEEE

clouds, our approach is to simply assemble single triangulated scans without triangulating the complete result. This of course is to be done offline in post process.

III. Implementation

A. Testbed Hardware

Even though the strong advantages of a fix pattern projector using an unknown pattern were pointed out in the chapters before, we are using a standard video beamer for projection purposes in the test bed. This enables us to easily switch between different resolutions and different patterns. If the preparatory work is done we will try out other projection methods, first of all using a gobo projector with strobe lighting.

The camera in the test setup is a standard digital video camera which is connected to a standard BT878-based frame grabber using the s-vhs input. The remaining system is assembled using a profile construction to allow easy adjustability in all directions. In the left corner of the base plate of the frame one can see the bracket that serves as a stopper for the acrylic calibration plates (fig. 3).

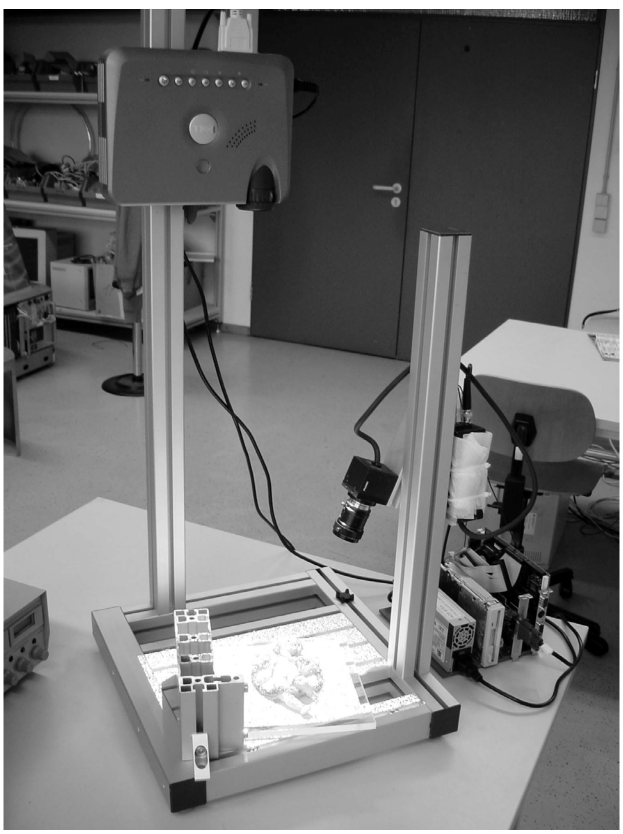

Fig. 3. Photo of the Testbed Hardware consisting of Standard Miniature Video Beamer on top and Camera on the right

Since we are still experimenting with different projection patterns at the moment, a small embedded pc is necessary. This is located at the right side of the test bed.

It is worth noting that it makes no difference whether the object is moved by hand under the video projector or the complete (miniaturized) scan unit is moved by hand over the object to scan.

B. Software Details

The algorithms as described above were implemented in C++ using the operating system Windows 2000. The Camera connection was established utilizing the video-for-windows (vfw) interface. The user interface was realized based on the windows API and based on an embedded vtk 3D render window for the resulting point clouds [13]. As described above, the 3D visualization will be changed towards OpenGL to allow faster data set refresh rates in the interactive process.

IV. Results

A. Examples and Screenshots

In first test runs of the system we tried to scan a variety of different objects, i. e. a textured box (a packet of cigarettes), a small goose figurine made of plaster, pieces of chalk, a computer mouse, and so on. Here we want to show a screenshot of the programs' user interface and of the resulting point cloud scanning a teeth plaster cast (fig. 4, 5). The speckle pattern is easy to recognize. We have chosen a relatively rough resolution of 320x240, which is compatible with the camera's resolution of 640x480, and to benefit from over-sampling in the pattern recognition procedure. Naturally the approach is easy to adapt to other resolutions using other hardware components.

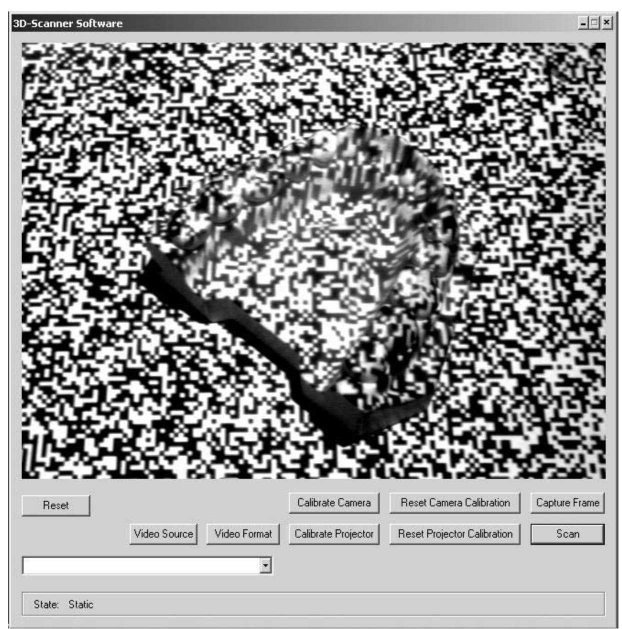

Fig. 4. Plaster Cast with projected Speckle Pattern

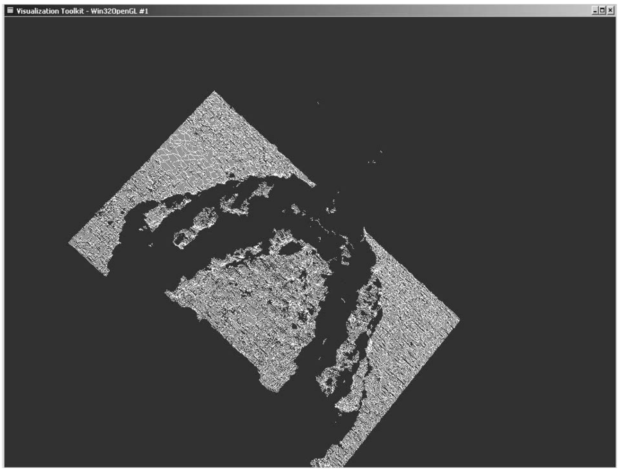

Fig. 5. Resulting Sample Scan of the Plaster Cast

B. Considerations on Accuracy

Due to the fact that the current implementation had only just been finished when this paper was written, we have not yet managed to perform intensive test runs. Anyway the calibration procedures deliver average errors of approx. 0.03mm for the camera calibration and approx. 0.08mm for the projector calibration. Of course this has to be put in relation to the chosen focal distance of the camera (zoom, approx. 12mm, CCD-to-object distance approx. 250mm, 2/3'' CCD).

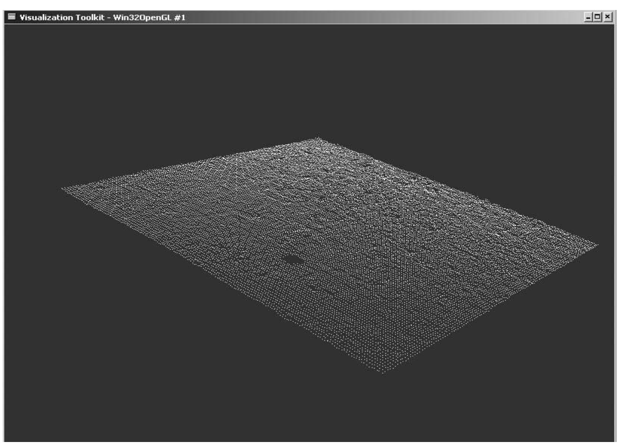

Fig. 6. Resulting point cloud of a plane scan for accuracy considerations

Another preliminary test was to calculate a regression plane through a point cloud of the scan of a plane, delivering a maximal deviation of approx. 0.51mm and a standard deviation in the range of the results delivered by the calibration procedures (fig. 6).

V. Conclusion and Future Work

At this point of time the algorithms described above are implemented in a test application and were proven to function correctly. The test-bed hardware consisting of video projector and digital camera on a frame assembly is easily scalable in geometry, resolution and focal distance. The next step will be to consider the system's errors and the maximal achievable accuracy in a theoretical manner. Furthermore we have to do a comparison of these results by intensive test runs, also with more complex objects than simple planes.

Another plan for the future is to replace the video projector with a strobe lighted gobo projector to minimize velocity distortions. Gobos have been already manufactured for this purpose and are available for use.

The fast icp registration procedure described is already implemented and it seems to work with test data sets, even given the tight timing constraints, but the final integration in the system is still to be done.

References

[1] H. Hoppe, S. Däuber, C. Kübler, J. Raczkowsky, H. Wörn, "A New, Accurate and Easy to Implement Camera and Video Projector Model", in *Medicine Meets Virtual Reality, Conf. 02/10 (MMVR)*, Newport Beach, January 2002, pp.204-206

[2] A. W. Gruen, "Adaptive Least Squares Correlation: A Powerful Image Matching Technique", in *South African Journal of Photogrammetry, Remote Sensing and Cartography, Vol. 14, No. 3*, 1985, pp. 175-187

[3] N. D'Apuzzo, "Modeling Human Faces with Multi-Image Photogrammetry", in *Three-Dimensional Image Capture and Applications V. Proc. of SPIE*, San Jose, California, 2002, Vol. 4661

[4] T. Luhmann, *Nahbereichsphotogrammetrie: Grundlagen, Methoden und Anwendungen.* Wichmann, Heidelberg / Germany, 2000 (german)

[5] S. Rusinkiewicz, O. Hall-Holt, M. Levoy, "Real-Time 3D Model Acquisition", in *Siggraph 2002*, San Antonio / Texas / USA, July 2002

[6] S. Rusinkiewicz, M. Levoy, "Efficient Variants of the ICP Algorithm", in *Proc. of the 3. Int. Conf. on 3D Digital Imaging and Modeling (3DIM)*, Quebec / Canada, 2001

[7] P. J. Besl, N. D. McKay, "A Method for Registration of 3-D Shapes", in *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 2,* February 1992, pp. 239-255

[8] F. Devernay, O. Bantiche, E. Coste-Maniere, "Structured light on dynamic scenes using standard stereoscopy algorithms", in *Institut National de Recherche en Informatique et en Automatique (INRIA), Research Report 4477*, June 2002

[9] P. Vuylsteke, A. Oosterlinck, "Range Image Acquisition with a Single Binary-Encoded Light Pattern", in *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 12, No. 2,* February 1990, pp. 148-164

[10] M. Maruyama, S. Abe, "Range Sensing by Projecting Multi-Slits with Random Cuts", in *International Workshop on Industrial Applications of Machine Intelligence and Vision (MIV-89)*, Tokyo, April 10-12, 1989, pp. 163-168

[11] M. Proesmans, L. J. Van Gool, A. J. Oosterlinck, "Active Acquisition of 3D Shape for Moving Objects", in *Int. Conf. on Image Processing*, Lausanne, September 1996, pp. 647-650

[12] E. Mouaddib, J. Batlle, J. Salvi, "Recent Progress in Structured Light in order to solve the Correspondence Problem in Stereo Vision", in *Proc. of the 1997 IEEE Int. Conf. on Robotics and Automation*, Albuquerque, New Mexico, April 1997, pp. 130-136

[13] vtk, visualization toolkit, Kitware Inc., free download via http://www.vtk.org