

Real-time 3D Reconstruction for Collision Avoidance in Interventional Environments

Alexander Ladikos, Selim Benhimane, and Nassir Navab

Technische Universität München

Abstract. With the increased presence of automated devices such as C-arms and medical robots and the introduction of a multitude of surgical tools, navigation systems and patient monitoring devices, collision avoidance has become an issue of practical value in interventional environments. In this paper, we present a real-time 3D reconstruction system for interventional environments which aims at predicting collisions by building a 3D representation of all the objects in the room. The 3D reconstruction is used to determine whether other objects are in the working volume of the device and to alert the medical staff before a collision occurs. In the case of C-arms, this allows faster rotational and angular movement which could for instance be used in 3D angiography to obtain a better reconstruction of contrasted vessels. The system also prevents staff to unknowingly enter the working volume of a device. This is of relevance in complex environments with many devices. The recovered 3D representation also opens the path to many new applications utilizing this data such as workflow analysis, 3D video generation or interventional room planning. To validate our claims, we performed several experiments with a real C-arm that show the validity of the approach. This system is currently being transferred to an interventional room in our university hospital.

1 Introduction

Today, it is not uncommon to have one or more automated devices, such as C-arms and medical robots, inside an interventional room. However, there is a danger of collision between these devices and other equipment such as ceiling-suspended monitor arms, radiation protection shields or the medical staff. A collision can damage both the device and the colliding object and sometimes requires to take the device out of service until a technician has evaluated the damage. The intervention may be suspended and the patient may be moved to a different room. Such collisions could be both dangerous and costly.

Currently, there is no fully automated solution to the problem of collision avoidance. Indeed, many systems rely on the discretion of the operating physician to avoid collisions. Technical measures include contact sensors and reduced movement speed. Some devices also perform a slow test run to check the trajectory for obstacles. The disadvantage of these mechanisms is that they only detect a collision when it has already occurred. Indeed, they are more focused

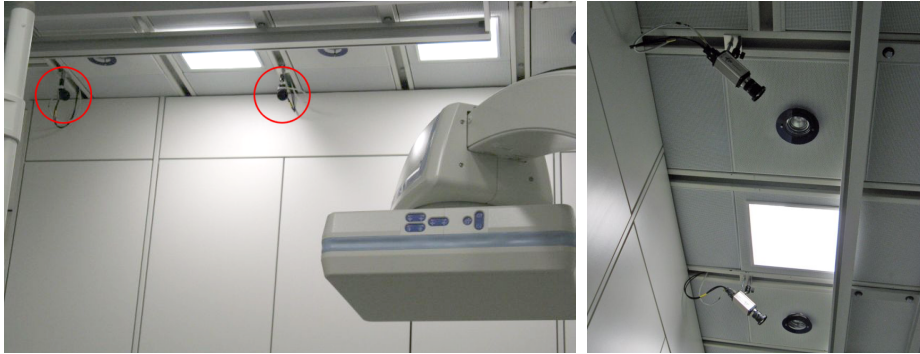


Fig. 1. Left: C-arm and two cameras (red circles) in our clinical installation. Right: Closeup of the cameras.

on minimizing the damage than on preventing the collision in the first place. In the case of C-arms, these safety considerations directly affect the speed of the device for certain movements. For instance, it has been shown in [7] that 3D rotational angiography can be performed in less time and with less contrast agent using angular C-arm motions instead of orbital motions. However, the C-arm speed required for this would cause serious injury in case of collision. With an automated collision avoidance, this method could be used more often and with increased safety for the medical staff. In recent years, medical robots [4, 2] have become an active research topic and some systems like the daVinci are regularly used in many hospitals. Sometimes such robots are also used in conjunction with other devices such as C-arms, navigation systems and electric tools. In such environments, staff unfamiliar with the movement range of all the devices can unknowingly enter the working volume of the device and cause an accident. This issue will become even more important in the future, since there will be many interventional rooms where multiple imaging modalities are used.

To overcome these problems, we propose a real-time collision avoidance system to increase the safety in the interventional room and allow faster device operation. Our system consists of 16 optical cameras mounted on the ceiling of the interventional room as well as a few PCs to perform the reconstruction and the collision avoidance (see figure 1). In an offline phase, the working volume of the device is determined and the cameras are calibrated. In addition, background images of the room are acquired. In the online phase, the background images are used to segment the foreground objects. The segmented images are subsequently used to perform the 3D reconstruction by intersecting the backprojections of the object silhouettes in space. In a final step, safety zones are computed around the objects and checked for intersections with the movement range of the device. If an object is inside the operating range of the device, a warning is given to the physician. The advantage of this method compared to existing approaches is that it is fully automated and detects possible collisions before they occur. It is also non-intrusive since the cameras are mounted on the ceiling and are

therefore not in the way of the clinical staff. This solution works for any kind of automated device. In addition, once the system has been set up, very little maintenance work is required. Our system meets the requirements for the presence of cameras in an interventional room since the images do not need to be saved. To the best of our knowledge, no previous work has considered a multi-camera-based collision avoidance system for an interventional room. Apart from collision avoidance, the system can also be used for other tasks such as workflow analysis, 3D video capture and interventional room planning.

2 Related Work

There is a vast body of work on 3D reconstruction in the computer vision literature. See [15] for a recent review. However, most work focuses on achieving high accuracy reconstructions using a large number of images without concern for the computational time. We are interested in real-time 3D reconstruction and will therefore restrict our discussion to this area.

Due to the real-time constraint, most systems only compute the visual hull [9] of the objects in the scene. One group of systems [12, 1] recovers the visual hull in a polyhedral representation [5]. The other group [3, 8] recovers the visual hull in a voxel representation [17]. The advantage of the polyhedral approach is that it computes an exact visual hull. The disadvantage is that it is not as robust to calibration and segmentation errors as the voxel-based approach, since it only relies on (accurate) silhouettes unlike the voxel-based approach which considers occupied and unoccupied regions. In the voxel-based approach the volume is discretized into voxels, which are tested for occupancy. This can be implemented efficiently and is very robust. However, the resolution of the visual hull is limited by the resolution of the voxel grid. In our system, we adopted the voxel-based approach, as it has the additional advantage that it can be directly used for the collision test. In addition, it allows us to limit the voxel resolution to the desired accuracy of the system thereby speeding up the computations.

3 Reconstruction System

3.1 System Architecture

Since the system is to be used in an everyday interventional setting, it is important that it is robust and non-intrusive. In addition, maintenance work should be limited to a minimum. These constraints are fulfilled by our system through several design decisions. First of all, the cameras are mounted on the ceiling. This takes them out of the way of the staff and prevents any unintentional movement of the cameras which would require a recalibration of the system. In addition, the placement on the ceiling prevents accidental occlusion of the cameras by other equipment which would reduce the robustness of the system. To ensure that all cameras take images at the same time, the cameras are externally triggered. The reconstruction is performed using four workstations and one master

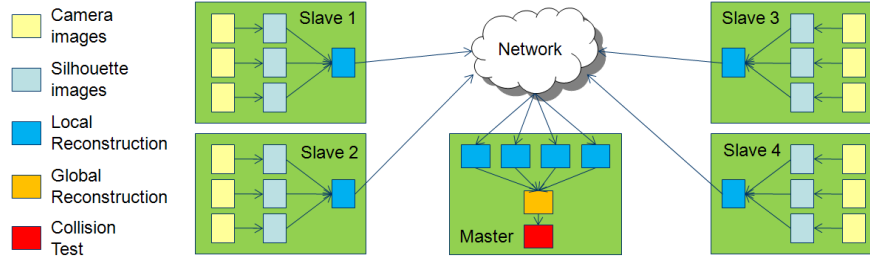


Fig. 2. Each workstation controls up to four cameras and is connected with the other PCs through a Gigabit Ethernet network. Each workstation performs a partial reconstruction using its locally attached cameras. The partial reconstructions are transmitted to the master PC which combines them and performs the collision test

PC. The workstations are equipped with a 2.6 GHz quad-core CPU, 2GB RAM and a 8800 GTX graphics board, while the master uses a 3.0 GHz dual-core CPU with 2GB RAM and a 8800 GTS graphics board. Figure 2 shows the data flow in the system.

3.2 Calibration

Before the system can be used, the cameras have to be calibrated and registered to the room coordinate system. The calibration procedure should be fast and easy to use, so that even a non-expert can safely perform it. We achieve this goal by using a method which does not use a calibration grid but instead relies on point correspondences created by a point light source such as an LED [16]. First, the lighting in the room is dimmed, so that it becomes easier to detect the light point. The user then has to move the light source through the reconstruction volume to create the correspondences used in the calibration. The points are automatically extracted from the images and used to compute the camera intrinsic and extrinsic parameters. In a final step, the camera coordinate system is registered to the room coordinate system using a calibration target at a known position in the room. The calibration procedure can typically be completed within 30 minutes.

3.3 Segmentation

The goal of the segmentation is to separate the foreground objects, which can collide with the device, from the static background objects. This is done using a robust background subtraction algorithm [6] working on the color images. Background images are recorded when the room is not in use. These images may only contain background objects, which are not part of the scene (i.e. no clinical staff or moving equipment). However, it is possible to have known occluding objects which cannot be removed from the room (i.e. the surgical table) in the background images. In section 3.5, we describe how to deal with this. During runtime, the current image is first corrected for illumination changes and then compared

to the background image. All pixels with a significant intensity difference are marked as foreground. Afterwards, morphological operations are used to filter small holes in the segmentation.

3.4 Reconstruction

The segmented camera images are used to reconstruct the visual hull of the objects in the room. To this end, the reconstruction volume is discretized into a voxel grid and each voxel is tested for occupancy by projecting it into each of the silhouette images. If the voxel projection is empty in one or more views, the voxel is considered to be empty for it is lying in unoccupied space. If, on the other hand, the voxel is occupied in all images, it is considered to be occupied. Assuming a perfect segmentation, this gives us a very strong guarantee on the reconstruction, namely that it will not miss any object in the room. As a matter of fact, this also holds true when certain regions are oversegmented because this can in the worst case only add occupied regions to the reconstruction.

To speed up the reconstruction, we implemented the occupancy test on the GPU. We precompute the voxel projections and store them in a lookup table. This has to be done only once when the system is installed since the camera configuration is static. Together with the distribution of the computations over multiple PCs, this allows us to run the reconstruction at real-time frame rates.

3.5 Occluders

One problem which has to be addressed is the presence of occluders in the scene. Occluders are objects inside the working volume which cannot be removed from the interventional room when taking the background images. The assumption during background subtraction, however, is that all foreground objects are located in front of the background. This is not the case in the presence of an occluder because a foreground object could move behind the occluder and effectively disappear from the segmentation. This will result in the partial or complete removal of the object from the reconstruction. This is unacceptable because it would make it impossible to prevent a collision between an occluded object and the device. To overcome this problem, we model the occluding objects, which are generally known, by either building or using an available CAD model. We can perform a 2D-3D registration using point correspondences between the images and the model to determine its position in the room. Using these models, we can test the visibility of every voxel with respect to the cameras. If a voxel is occluded in a camera, the corresponding region in the segmented images is not considered during the reconstruction.

3.6 Collision Test

During runtime, a bounding box is created around every object in the reconstruction and checked for an intersection with the device movement range. Our

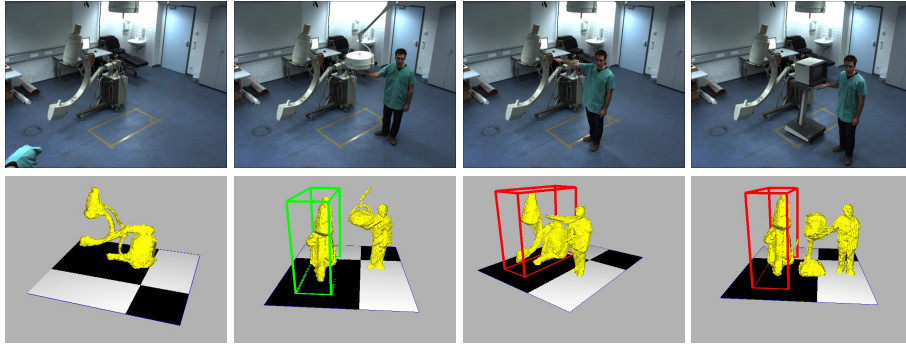


Fig. 3. Results of the collision avoidance with a C-arm. Each column shows one of the input images in the upper row and the reconstruction in the bottom row. The first column shows the reconstructed C-arm. The second column shows the C-arm in a safe state (green bounding box), while the third and fourth column contain an object in the safety zone of the C-arm (red bounding box).

system is not limited to the use of bounding boxes. Since we reconstruct the shape of the objects, we can also use their actual shape for the collision detection. If there is an intersection, the physician is notified visually and acoustically. To aid the physician in quickly finding the responsible object, we visualize the reconstruction after applying the Marching Cubes algorithm to obtain a mesh representation. The system is only active when the device is being operated, so that the physician is not disturbed by alarms, when it is safe to enter the working volume of the device.

4 Results

We have two setups of the system. One is in our lab and the other is in a permanent installation in an interventional room in our university hospital. The experiments were performed with the lab system because the system in the hospital is not fully installed yet.

Our working environment has dimensions of size $3.7 \times 3.2 \times 2.2$ m. We use color cameras with a resolution of 1024×768 pixels which are externally triggered at a frame rate of 30 fps. The working volume is discretized into voxels of 2.2 cm side length which is therefore the accuracy of the system. This is sufficient in practice since we would like to maintain a security distance of at least 20 cm to the device. The experiments were performed with a C-arm. During runtime, we compute the bounding box for each object in the scene and test it for intersection with the safety zone of the C-arm, which is modeled as a rectangular working volume that extends 20 cm beyond the C-arm bounding box. If there is an intersection, a warning is displayed (bounding box turns red) and an alarm is sounded. Figure 3 shows some reconstruction results and some configurations with objects inside

and outside the safety zone. All intersections with the safety zone of the C-arm were successfully detected. The segmentation and the reconstruction on one workstation take each approximately 15 ms, while the data transmission takes about 10 ms. The computations are pipelined, allowing a frame rate of 30 fps.

5 Discussion

Our system is very welcome by our clinical partners. That is why we were able to install it in a real interventional room. Once the installation is complete, we plan to perform extensive experiments under real interventional conditions. Judging from our experimental results, we will need to adapt the segmentation algorithm to the colors and the illumination changes in the interventional room. This will avoid undersegmentation and prevent holes from appearing in the reconstruction. The current number and position of cameras will also need to be adapted to the layout of the interventional room in order to achieve optimal reconstruction results. In addition, we plan to use point-to-point distances between the objects and the device to determine collisions.

5.1 Other Application Areas

Workflow Analysis Workflow analysis deals with recovering and analyzing the workflow of an intervention [11, 10, 14]. For instance, it can be used to document the surgery and to compare the performance of different surgeons. Our 3D reconstruction system allows these methods to use a new modality, namely the shape and the position of objects in the scene, as an additional input.

3D Video Texturing the reconstruction using the input images, allows us to obtain a 3D video [13] of the intervention if regulations allow it. Contrary to a regular video recording the viewer can choose his viewpoint freely, giving him more insight into the action. This can be used for documentation, analysis and training. In addition, by analyzing the movement of the personnel during the intervention in 3D it is possible to improve the interventional room design.

6 Conclusion

We presented a real-time 3D reconstruction system for collision avoidance with automated devices in an interventional room. This creates a more efficient working environment. The anti-collision system can be used to increase the speed of the device, allowing better reconstruction results in the case of C-arms. In addition it prevents medical staff to inadvertently enter the working volume of the device. The system is non-intrusive and requires little maintenance. The recovered 3D reconstruction is also useful for other application areas such as workflow analysis and 3D video. Our future work will include performing extensive experiments and validations in the real interventional room once the system has been fully installed.

Acknowledgment: This research is funded by Siemens Medical Solutions, Forchheim, Germany. We would also like to thank Dr. Thomas Redel and Dr. Klaus Klingenberg-Regn for their support.

The concepts and information presented in this paper are based on research and are not commercially available.

References

1. J. Allard, C. Menier, B. Raffin, E. Boyer, and F. Faure. Grimage: Markerless 3d interactions. In *SIGGRAPH - Emerging Technologies*, 2007.
2. D. Asai, S. Katopo, J. Arata, S. Warisawa, M. Mitsuishi, A. Morita, S. Sora, T. Kirino, and R. Mochizuki. Micro-neurosurgical system in the deep surgical field. In *MICCAI*, pages 33–40, 2004.
3. G. Cheung, T. Kanade, J. Y. Bouguet, and M. Holler. A real-time system for robust 3d voxel reconstruction of human motions. In *CVPR*, 2000.
4. G. Fichtinger, J. Fiene, C. Kennedy, G. Kronreif, I. Iordachita, D. Song, E. Burdette, and P. Kazanzides. Robotic assistance for ultrasound guided prostate brachytherapy. In *MICCAI*, pages 119–127, 2007.
5. J. S. Franco and E. Boyer. Exact polyhedral visual hulls. In *BMVC*, 2003.
6. S. Fukui, Y. Iwahori, H. Itoh, H. Kawanaka, and R. Woodham. Robust background subtraction for quick illumination changes. In *PSIVT*, 2006.
7. J.-Y. Gauvrita, X. Leclerc, M. Vermandelb, B. Lubicza, D. Despretzd, J.-P. Lejeunec, J. Rousseaub, and J.-P. Pruvoo. 3d rotational angiography: Use of propeller rotation for the evaluation of intracranial aneurysms. *American Journal of Neuroradiology*, 26:163–165, 2005.
8. J. M. Hasenfratz, M. Lapierre, and F. Sillion. A real-time system for full body interaction with virtual worlds. *Eurographics Symposium on Virtual Environments*, pages 147–156, 2004.
9. A. Laurentini. The visual hull concept for silhouette-based image understanding. *PAMI*, 16(2):150–162, 1994.
10. J. Leong, M. Nicolaou, L. Atallah, G. Mylonas, A. Darzi, and G.-Z. Yang. Hm assessment of quality of movement trajectory in laparoscopic surgery. In *MICCAI*, pages 752–759, 2006.
11. H. Lin, I. Shafran, T. Murphy, A. Okamura, D. Yuh, and G. Hager. Automatic detection and segmentation of robot-assisted surgical motions. In *MICCAI*, pages 802–810, 2005.
12. W. Matusik, C. Buehler, and L. McMillan. Polyhedral visual hulls for real-time rendering. In *Eurographics Workshop on Rendering*, 2001.
13. T. Matsuyama and T. Takai. Generation, visualization, and editing of 3d video. In *3DPVT*, 2002.
14. N. Padoy, T. Blum, I. Essa, H. Feussner, M.-O. Berger, and N. Navab. A boosted segmentation method for surgical workflow analysis. In *MICCAI*, pages 102–109, 2007.
15. S. Seitz, B. Curles, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR*, 2006.
16. T. Svoboda, D. Martinec, and T. Pajdla. A convenient multi-camera self-calibration for virtual environments. *Presence: Teleoperators and Virtual Environments*, 14(4):407–422, 2005.
17. R. Szeliski. Rapid octree construction from image sequences. *CVGIP: Image Understanding*, 58(1):23–32, 1993.