Technische Universität München
Institut für Informatik

Diplomarbeit

High Accuracy Tracking for Medical Augmented Reality (Hochgenaues Tracking für Erweiterte Realität in der Medizin)

Tobias Sielhorst

Aufgabenstellerin: Prof. Gudrun Klinker, Ph.D.
Betreuer: Ali Khamene, Ph.D.
Abgabedatum: 24.10.2003

Ich versichere, dass ich diese Diplomarbeit selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

Special thanks to Ali, André and Susanne

Abstract

Augmented Reality is an emerging technology combining virtual reality media with perception of reality in order to present information in a very intuitive way. This young technology offers a wide range of use from supporting any professionals in performing complex actions to mere entertainment. One of the major problems is tracking the user's position for projecting virtual objects in the expected place. There are several different tracking techniques with different advantages and drawbacks. In the environment of medical Augmented Reality a way of tracking is needed that yields high and dependable accuracy and robustness while a large spatial range may be neglected. For this purpose the RAMP[10] Project at Siemens Corporate Research takes advantage of visual inside-out tracking of a single camera. In this thesis it will be shown how the needs can be met. Instead of tracking natural markers in images, artificial landmarks are introduced into the setup. By choosing a very simple shape for these fiducials the image has less complexity which enhances robustness of the algorithms and allows for fast computation time. Fast calculation allows for adding more redundancy for more accuracy. As a drawback of the simple fiducials new algorithms had to be created in order to extract implicit information about visibility, partial occlusion and identification of each fiducial.

Contents

1 Introduction

Computers play a central role in information society. Rapid development in computer technology in the last eight decades from the first digital programmable computer of Konrad Zuse to the integrated processors with millions of transistors connected with numerous other computers in networks offer more and more information. Therefore, the question how to provide an efficient interface for human-machine communication gets more and more important. A general development is the fact that computers have to provide their data in a more human way because it is necessary to display more complex data in a more understandable way. In contrast to abstract figures and numbers our brain can handle easily our normal environment or anything that behaves like it. The ultimate closeness to this aim would be virtual objects behaving like real objects, placed seamlessly into our environment.

1.1 Augmented Reality

Augmented Reality (AR) is an emerging technology combining reality with virtual objects. Milgram and Koshino [44] define AR by their Reality-Virtuality-Continuum as a Mixed Reality that emphasizes the reality as the prevailing environment into which virtual objects are inserted (see figure 1). In contrast to Virtual Reality a full immersion into a virtual environment is not desired. The aim is integration of virtual objects into the real environment. A major difference between Augmented Reality and Virtual Reality is the fact that not all of the environment has to be modeled nor known [37].

Azuma [16] defines Augmented Reality by three characteristics:

1. **Combines virtual and real**.
2. **Interactive in real time**.
3. **Registered in 3D**.

This means that Augmented Reality is not limited to visual augmentation even though the major part of research focuses on visual augmentation [15]. This development can be explained by the complexity of data that can be displayed by visualization. A proverb claims that 'An image is worth a thousand words' addressing the complexity of information that can be transferred at a time. Scientists confirm the commonly accepted opinion that the sense of vision is the most complex and developed one of or our senses [50]. Still, augmentation has been applied for aural [47] and haptic [72] senses as well.

Augmentation does not only consists of adding objects, but of removing them as well. A technology that can add objects can potentially remove

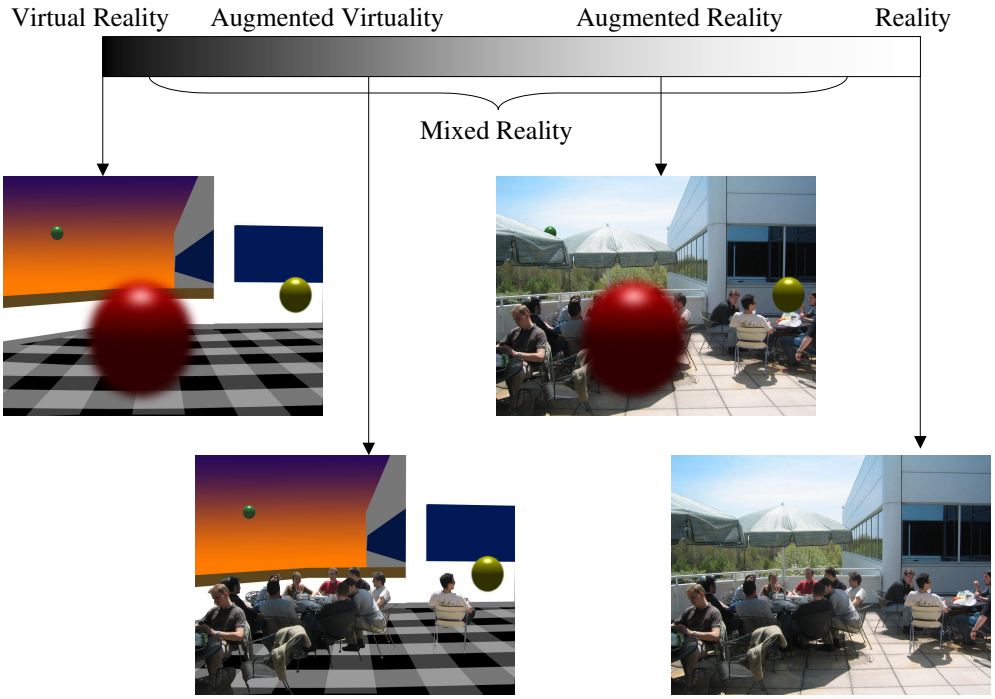

Figure 1: The Reality- Virtuality- Continuum

objects by superimposing a background e.g. a wall. This gives the impression of transparency of real objects (see figure 2).

The term Augmented Reality dates back to 1992 when the work of Caudell and Mizell [52] was published. The system was supposed to support mechanics in maintaining and assembling airplanes. Unfortunately, the project did not end up in commercial use because of too many open questions but it inspired many.

Since 1998 Augmented Reality enjoys its own international IEEE workshop IWAR [11], [12] or symposium ISAR [13] (leading to ISMAR [14],[15] as a symposium for Augmented Reality as well as for Mixed Reality) illustrating the interest of research.

1.2 Medical Augmented Reality

Many of us know the first visions of medical Augmented Reality from TV. Whenever doctor McCoy from the series "Star Trek" took his fancy device and held it over the patient and the device displayed a broken bone, we could see medical Augmented Reality at work. As the automatically opening doors

Figure 2: Removal of a table and two people by overlaying the background

that stunned many viewers thirty years ago and that are pretty common at shops nowadays, Augmented Reality is heading from future to present.

Robinett suggested in 1991 to use registered ultrasonography combined with virtual reality devices for what he called a little flowerily 'X-ray view' which can be said to be the birth of medical Augmented Reality. One of the first trials to implement medical AR together with ultrasonography was made in 1992 [74].

Augmented Reality can be applied for many different tasks such as architecture and construction [68], air traffic control [73], industrial maintenance and assembly [69], mining, archeology [70], military training [78], and entertainment [46]. Each application has its own special needs for accuracy, latency, range of use, robustness, costs, preparation of environment, and weight. These requirements lead to very different systems even though all systems aim to present virtual objects in a real environment. Medical Augmented Reality has the following constraints:

1. **High accuracy**.

2. **Dependable accuracy**.
3. **Robustness**.
4. **The advantage of using the AR system must rectify its price**.

Accuracy is the main emphasis in medical Augmented Reality. If a doctor has to decide on a therapy or make an incision based on the position of virtual objects the system must be accurate and this accuracy must be dependable. Doctors' decisions are affecting the future health of patients, so the technology is not allowed to fail. An error level that might mislead must either be detected or not occur at all. For this reason the maximum error must be controlled while the average error is unimportant. Note that high accuracy and dependable accuracy are two different issues . The maximum error being below a certain limit means dependability while a low limit stands for high accuracy.

Robustness supports dependability by different means than accuracy. Since doctors must trust in the reliability of an AR system, it must look reliable as well. That means AR will only be used for medical purposes if the system *is* reliable and *appears* to be reliable as well. Therefore the system must be robust. If the system stops the augmentation for a few frames it might not harm anyone, but the doctor loses trust. As an analogy, the same happens if an elevator is stuck. This incident makes the person inside think that using an elevator is not safe although the emergency brakes have done their job (in most cases overeagerly) to prevent the elevator from falling down in any case. Thus, the person loses trust in the technology he cannot control himself. Of course, the patient must have faith in the system as well, but this cannot be helped by technology. This is an ethical problem.

Last but not least there is the delicate topic costs. We must keep in mind that Augmented Reality is a supporting technology, not a treatment. Hence, the price of an AR system will always be on top of the treatment itself. Research about medical AR must face the fact that a high price limits the possibilities of a system. A costly AR system is unlikely to be combined with an inexpensive modality like ultrasonography for instance. This modality is wide-spread because of its reasonable price. The benefit from augmention must be very high in this combination to become a real application. Therefore the price is also an important constraint. One may object that any developed AR system has a strong cost constraint but e.g. at military the cost constraint might be less strict. In military a strategic advantage due to AR might be worth a lot and furthermore the costs of the whole technology like e.g. a jet exceeds the costs of the AR part by far. In that field, costs are of secondary importance. As another example, AR in

entertainment may benefit from mass production and expensive parts may be exchanged with ones of lesser quality. In entertainment the costs can be a trade-off to quality while at medical AR the cheapest technology with the desired accuracy and robustness has to be found.

On the other hand there are benevolent properties of medical Augmented Reality that can be exploited to maintain the critical demands of the system. These are requirements of fields of application other than medicine that can be neglected. The environment can be prepared in advance since the system probably stays in the same room. Hence, hardware to support tracking (see section 3) can be installed into the room of usage. Furthermore the system's weight is unimportant since the computer can stay in the same place and need not be worn necessarily. The range of use can be limited since the patient is generally not moving too much during these procedures. The doctor usually stays at the same side of the patient as well, so the spatial range of use of the AR system can be limited to a minimum.

2 Human visual perception

> People believe eyes more than ears;
> it is a long way by instructions, short and effective by examples.
> *Seneca.*

This Diplomarbeit is about visual tracking for visual augmention. Thus, there are two good reasons for having a closer look at human visual perception. If we want to model virtual objects that make us believe they behave like real objects we need to know about the physics of the objects as well as our own visual perception. Understanding our visual system is the key to useful augmention.

Pose estimation is done by our eyes together with the brain all the time in a robust and fast way, so it might be a wise idea to be inspired by the human visual system. Millions of years of evolution made our eyes and our brain to an impressive visual tracking system we might learn from a lot.

But first of all: Why vision?
It is in our nature to perceive information mainly visually. 'Complex three-dimensional relationships, procedures and constructions can be presented by computers in form of artificial worlds. These offer users access to three-dimensional information by using our highly developed visual sense of perception' [36]. Also psychologists agree that the sense of vision is the most highly developed sense and the most important one [50]. That makes vision the first choice for augmention.

2.1 The eye

The actual organ of vision is the eye that consists of the eye ball, eyelids, lachrymal and eye muscles. The striking similarity of eyes and cameras is of course no coincidence, since the camera is build to imitate the human eye. Both use a lens in order to focus incoming light. To achieve a sharp image the lens must be accommodated correctly according to the distance of an object. In contrast to the eye not all cameras have an auto focus system. The brain uses the depth of accommodation as a cue (see 2.2) for estimating the distance. Accommodation is performed by reflex. It adjusts the lens refraction by using muscles attached to the lens. It aims to obtain high frequencies in the fovea.

Diaphragms in cameras and the iris in the human eye control the quantity of light hitting the photo sensors or receptors respectively. Reducing the incoming light by closing the aperture or pupil causes a higher depth of

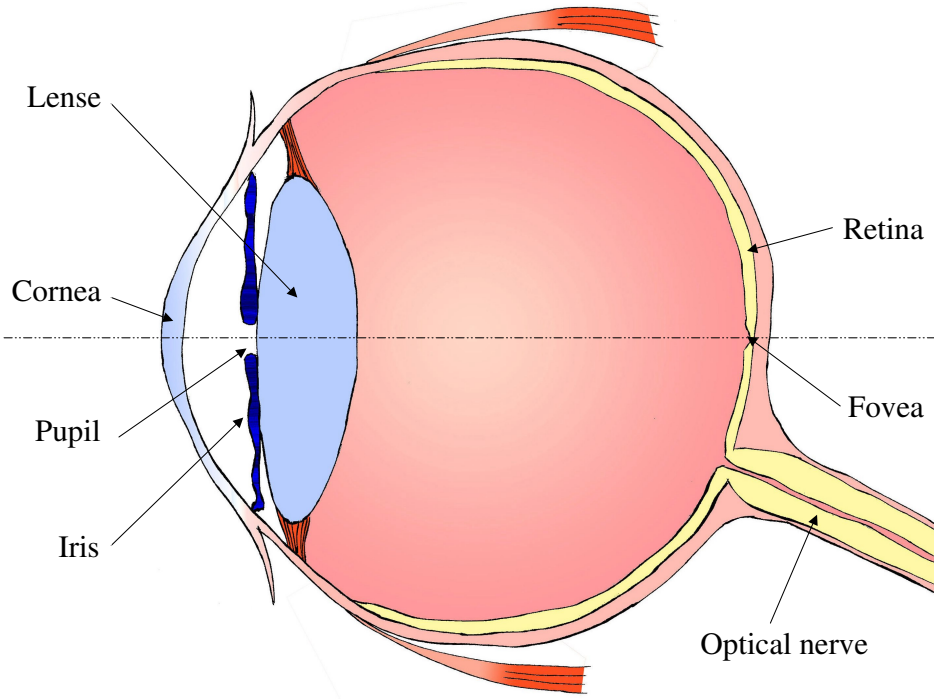

Figure 3: Sargittal cut through the eye ball

sharpness because the optics behave more and more like a pin hole camera that projects any object at any distance sharply.

There are also differences in the architecture of the eye and its technical pendant. While a camera has flat projection screen, the one of the eye is ellipsoidal. That is why the the resolution of the eye is given in angular dimension and the resolution of a camera is given in millimeters at a certain distance. The resolution of the retina is completely different from any camera. Ordinary cameras have a linear resolution that is the same on the whole sensor chip while the resolution of the eye is highest in the fovea at 1/60 degree and decreases with the distance from the optical axis. This resolution makes normal sighted people able to see objects as small as 1.5 millimeters at 5 meter distance. The retina responds to light logarithmically. That means the difference between a 25 Watts and a 50 Watts light bulb is perceived like the difference between a 50 Watts and a 100 Watts light bulb. The range is 14 logarithmic units which means if 1 is the lowest unit of luminosity to respond to. 10^{14} is the highest luminosity and it might harm the retina. The photo receptors can be divided into two kinds. About 6 million cones respond to light of wave length of either red, blue or green light. Therefore,

mixes of different wavelengths of light can only be perceived as a mix of these three colors. Cones have a high acuity and they can mostly be found close to the optical axis. The other kind of receptor is rods that respond to any visible light. Rods are much more numerous than cones with about 120 million rods in each eye. They are much more sensitive to light but they cannot distinguish between color. That is why we cannot see colors in the dark. Several rods are connected through a single nerve. This increases the chance of activation of this nerve at low luminosity. Rods can detect motion better than cones but they offer less acuity. This and further information about human vision can obtained at [51], [50] in German and [84], [83] as English online textbooks.

2.2 Depth perception

Vision implies the loss of one dimension. The three dimensions of space are reduced to a two-dimensional image that the brain tries to understand. This kind of introduction to the topic might look like the end of this section, but fortunately, after projection from three dimensions to two, there is much three-dimensional information remaining in the images. These pieces of information are called depth cues because they might be inaccurate, ambiguous or even contradictory. Drascic and Milgram [37] present an overview of visual issues in Mixed Reality in general and Augmented Reality in particular. The main interest of that paper is depth perception. A short summary of depth cues used by human perception is shown in table 1. Fusion of these cues is not trivial, especially if they are contradictory. How the brain computes a model of the three-dimensional space from the information is still disputed. There are four theories commonly presented [38] as possible explanations how depth cues might interact with each other.

1. **Accumulation or Weak Fusion**. The visual system estimates the depth from each cue separately and takes them into account as a kind of linear combination.

2. **Cooperation or Strong Fusion**. Cues cooperate before a depth estimation is done. Depending on the reliability of each cue, its relative importance varies in each situation.

3. **Disambiguation**. One cue solves an ambiguity of another cue. Shading for instance can provide ambiguous estimates since a hollow half sphere has the same shading from the inside as from the outside. This ambiguity can be solved by stereo vision while shading provides a more detailed depth model.

Pictorial Depth Cues	Occlusion or Interposition
	Linear Perspective
	Shadows
	Texture (Detail) Perspective
	Aerial Atmospherical Perspective
	Relative Brightness
Kinetic Depth Cues	Relative Motion Parallax
	Motion Perspective
	Kinetic Depth Effect
Physiological Depth Cues	Convergence
	Accommodation
	Blur
Binocular Disparity	Stereo Disparity

Table 1: Depth cues

4. **Veto**. Cues can veto one another if they are in conflict. This phenomenon cannot be explained by simple weighting of each cue. Strong cues might not be challenged at all when in conflict with weaker ones. Weaker ones are simply ignored in that case. As an example there might be many pictorial depth cues in a photo but no normal-sighted person would doubt that a photo is flat.

There is strong evidence for each theory depending on the combination of depth cues. Thus, fusion does not only depend on the importance of the cue but also on the content and the special interaction of the combination of two particular cues [79]. There is no obvious rule which one of these four fusion techniques is applied. While convergence has more importance than accommodation to our brain, it weights these cues together to an obviously wrong result in the hope of being closer to reality. This is along with wrong brightness one of the reasons why objects augmented with an optical see-through HMD[6] are perceived at a wrong distance. Stereo vision and shading depth cues are computed together differently. Stereo vision simply vetoes results from shape-from-shade cues. This means that a flat disc with the same shading as a sphere is still perceived as a disc. Watching it with only one eye it appears to be a sphere. We are not conscious about these processes and they are working well because they are custom made for our normal environment. But what if our perceptual system is confused by artificial information that is partly misleading our senses?

2.3 Cyber sickness

There are three ways of responding to confusing data which can occur at the same time: Managing, adapting and refusing. The first one is the simple use of the perceptual system as described above even though we know that the information is inaccurate. According to [38] and [37] we are fast at adapting and learning to correct wrong estimates by other cues and senses. Tests with contradictory cues were taken with better results by experienced persons than by persons doing the test for first time. Refusal can be a little bewildering like when looking at M.C. Escher's famous staircases. But it can also become physically sickening. This phenomenon is called cyber sickness or simulator sickness. It is an inaccurately named syndrome (but not sickness) with the symptoms of vertigo, nausea, sweating, eye strain, fatigue, blurred vision or disorientation. It can occur when our visual, vestibular[13] and proprioceptive[8] senses collect contradictory data.

Kolasinski [39] names as much as 39 different factors for simulator sickness ranging from individual factors (like age, mental rotation ability or experience with simulators) and over simulator induced factors (e.g. flicker, contrast, scene content or time lag) to task factors (e.g. self movement speed, sitting versus standing, degree of control). In 1993, tests [39] have been performed to find out how many people suffer from this kind of sickness with an off-the-shelf virtual reality system of that time. A mere 39% of the subjects stayed without any sypmtoms. This means that more than a half of the partpiciants suffered from simulator sickness.

An interesting question is whether the same well explored results from Virtual Reality can be applied for Augmented Reality. This depends on which theory on cyber sickness is correct. There are two different [39] theories to explain the occurrence of cyber sickness

- **Cue Conflict**. Contradictory cues lead directly to sickness. It is the oldest theory on cyber sickness and widely accepted. Critics note that it has too little predictive power. In some cases people do not get cyber sickness although they experienced severe contradictory stimuli. It just states that in a case of cyber sickness conflicting cues must have occurred.

- **Postural Instability**. A more powerful theory states that cyber sickness is caused by postural instability. It is more powerful because it explains situations which are exceptions to the other theory. Postural instability can be caused by contradictory cues but it claims that if there are enough correct cues to maintain postural stability no cyber

sickness will occur. The same applies if the subject is used to the situation and the brain can pick the correct cue.

None of these theories prevailed against the other recently and none of them gives a comprehensive model of cyber sickness. Note that both theories state that contradictory cues may lead to cyber sickness, but there is a slight difference for deriving results from Virtual Reality to Augmented Reality. If the first theory is correct Augmented Reality causes with the same technology more frequently cyber sickness because according to Milgram and Drascic [37], coherent depth cues are easier to maintain in Virtual Reality (or: 'stereo graphics' as they refer to it) than in AR (or: 'combination of stereo view and stereo graphics') because of additional alignment errors. If the second theory turns out to be correct Augmented Reality provides more information about the real world supporting postural stability than Virtual Reality, so AR would be expected to cause less cyber sickness. On one statement both theories agree: Reducing contradictory cues decreases the chance of cyber sickness.

There is also a theory on why the body reacts with cyber sickness. Kolasinski [39] quotes the theory that the sense of balance might serve as a very sensitive indicator for the intake of poisonous food. The body does not expect cues from the sense of balance that cannot occur in an environment without technology. Since physics do not change, the nerve cells are expected to measure incorrectly which might be caused by poisonous food.

A strategy against cyber sickness would be attacking any of the factors inducing cyber sickness mentioned above, but we cannot reduce personal factors except by training. Task-induced factors can only be limited. Thus, strong effort must be made in order to keep technical factors for cyber sickness as low as possible. Augmented Reality will not be accepted by professionals before its sickening side effects are reduced to a minimum.

3 Tracking

According to its definition (see 1.1) augmented Reality shows virtual objects registered together with real environment. An Augmented Reality (AR) system must at least gain information about the user's head relative to the virtual objects in order to show the objects in the expected spot. In every AR system pose estimation is done continuously and therefore it is mostly called tracking. Tracking in AR need not mean mere pose estimation of the user's head but also pose estimation of tools or other objects in the environment.

3.1 Introduction to different kinds of tracking technology

There are different kinds of technologies to find out the position of an object. An overview with more detailed information can be obtained in [57]

- **Magnetic tracking**. It works in principle like a compass. The tracking device measures the direction of a magnetic field. This can either be the earth's magnetic field to obtain only the orientation or it can be an artificially generated magnetic field with a number of sensors for an exact three-dimensional positioning. Magnetic tracking yields high accuracy with a high update rate and robustness. As disadvantages, there are a short range of use and strong distortions if ferromagnetic material is in the vicinity.

- **Global satellite-based positioning systems**. NAVSTAR/GPS (USA) [90], Glonass (Russia) and soon Galileo (EU) [91] are outdoor tracking systems calculating the distance to at least three orbiting satellites. Although it is very popular outdoors because of its availability everywhere, it has no importance in medical Augmented Reality because is is not available in buildings and the accuracy is in the range of meters but not millimeters and below.

- **Connected joints**. This is a mechanical way of tracking where the object to be tracked is physically linked to the point of reference. It yields high accuracy and a high update rate. For Augmented Reality is is very impractical because the object to track would be the HMD which would have to be physically linked to a referencing object. This kind of tracking is used if the physical link exists anyway as in a robot.

- **Inertial tracking**. This kind of tracking imitates the proprioceptive[8] sense. The accelerometer measures its acceleration. The gyroscope

measures the speed of rotation. The relative position to the starting point can be obtained by integration in combination of both. Inertial tracking has a high update rate and it is self contained, i.e. there is no need of additional transmitters or fiducials. Unfortunately its error is accumulating over time. Additionally, accurate accelerometers are rather expensive.

- **Visual tracking**. This kind of tracking imitates human visual pose estimation. The position of an object is estimated by computation from images with the object in it. Visual tracking yields supreme accuracy, it is relatively inexpensive and it may have a large area of use. On the other hand it is computationally costly and it lacks robustness because tracked objects must be visible. Due to its high potential in accuracy the whole of section 3.2 is dedicated to this kind of tracking. Examples for pure visual tracking in AR are [59] and [45].

- **Hybrid tracking**. Since all presented ways of tracking have their advantages and drawbacks, it is an obvious approach to combine at least two types of tracking to gain from both. Examples are magnetic and visual tracking [53], [58] or inertial and visual tracking [64], [54].

There are also other tracking technologies like acoustic tracking (sonic depth finder) or tag trackers we know from theft protection in shops, GSM location etc. However, the itemized ones above play the most vital role in Augmented Reality.

3.2 Visual tracking

Visual tracking yields high accuracy and is inexpensive. Its mathematical background seems to be well understood while its robust implementation is difficult [71]. There are different design decisions to be made in visual tracking.

3.2.1 Stereo vision versus one camera vision

It is a very natural approach to use more than one camera for tracking. Most animals have two eyes in order to explore their environment. Human perception makes use of stereo vision in order to obtain three-dimensional information. With a multiple camera view, depth can be computed by triangulation without further knowledge about the objects such as size or shape. For accurate results the cameras must have a considerable distance. Figure 4 demonstrates the problem. Stereo camera systems have, of course, a higher

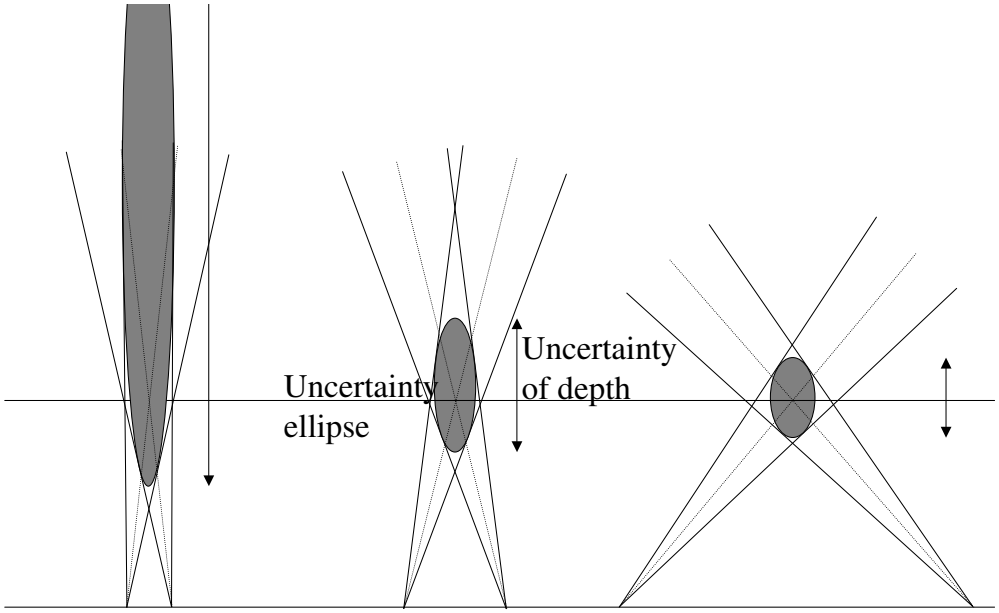

Figure 4: This figure illustrates the impact of the distance between two cameras in a stereo camera system on accuracy of depth. Note the difference sizes of uncertainty ellipses of three different camera distances with the same convergence distance and the same deviation in error.

chance of one of their camera views being occluded than a single camera system. This is an issue especially for medical Augmented Reality. In this environment other people as well as machines and tools are present.

As nature shows us in human depth perception (see section 2.2), stereo vision or multiple camera vision is not the only way of perceiving depth. Especially if there is knowledge available about the objects in the image, accurate depth estimation is feasible. Only four points on a rigid body with known relative coordinates must be visible in a single image to obtain the relative position to the camera (see section 4.7).

3.2.2 Natural landmarks versus fiducials

Our eyes continuously extract three-dimensional information from our normal environment without any aids for easier image processing. Even though the theory behind self-calibration seems to be well-understood [71], computers (or at least their programmers in this era) have a hard time extracting accurate

spatial information from unprepared environments. Techniques for extracting estimations using unprepared environments are not only too inaccurate for medical procedures, but they are nowadays still too slow as in [55], [26], [5], and [70]. Additionally, visual trackers can only determine a relative movement from where the system has been started, except if certain objects can be recognized which is a tricky task if the objects are not prepared with markers. Therefore many Augmented Reality systems employing visual tracking use fiducials[4]. With the knowledge about their nature, algorithms for exact feature extraction can be adjusted to the fiducial. Of course, the fiducials can be made in a way that algorithms can be kept as fast and robust as possible. Furthermore, a fiducial can reveal its absolute position if measured beforehand, or identify the object it is attached to. Therefore fiducial-based visual tracking is the first choice if the environment can be prepared easily.

3.2.3 Active fiducials versus passive fiducials

Fiducials[4] can be roughly divided into two groups. While active markers as used in [66] emit light, passive markers as in [53], [69], and [67] simply reflect it. Active markers usually stand out against the background of an image by brightness. Their contrast to the background can be made as high as desired. Hence, image processing is easy and a insufficient illumination is not a problem. On the other hand active markers are more difficult to handle. They are dependent on a source of power. Usually this is a battery adding weight and increasing the chance of malfunction since the battery can be empty. Technical malfunction is impossible with passive markers. No diode, bulb nor circuit can be broken. They can be made of any material that is compatible with the setup while active markers have metal inside for the technology. Passive markers are also popular because they are generally less expensive and a way of obtaining them might be a simple printout as in [24], [30], and [43]. Passive markers are all in all easier to handle but image processing can be more tricky because their images may have too little contrast compared to the background and they may suffer from shading and shadows.

3.2.4 Controlling illumination

Except in an environment with active markers, image processing can be disturbed by bad illumination. As usual problems there are too little illumination, shadows and shading. In case of too little light aperture time can be increased in photography. In Augmented Reality it is limited to the update

rate of the system. If the update rate of the visual tracker is e.g. 30Hz the shutter time cannot exceed $1/30s$ either. Additionally, a shutter time faster than the update rate would be desirable to avoid motion blur.

Of course, the accuracy of measurements depends on signal-to-noise-ratio. If it drops because of underexposure, the measurement from the images will deteriorate, too. Within a close range the problem of too little illumination can be solved by a flash. If this flash is very close to the camera shadows can be eliminated as well. The same light that helps computer vision might disturb the user, coworkers of the user and other cameras. Especially in a medical environment this is an important issue. Therefore a common technique is to perform tracking in the infrared spectrum. In this spectrum, the environment can be illuminated without user's notice. CCD photo sensors in video cameras respond also to infrared light in contrast to human eyes. Usually video cameras have a filter in front of the objective to absorb infrared light (∼940 nm). Hence, this one has to be replaced by a filter absorbing only light in the visible spectrum (below 820 nm) [67]. This approach can be improved by using retro-reflective material for the object to track. This kind of measure to control lighting conditions with an infrared flash and retro-reflective material has been used in [67], [77], in RAMP[10] and other AR systems e.g [85], [86].

If illumination is not controlled lighting conditions may change, too. Different lighting conditions need algorithms that adapt to them, which takes up computational time which is sparse in real time systems. Furthermore if we control the lighting conditions we can emphasize the objects we are interested in. This reduces the complexity of the image and results in a faster or more robust performance of the algorithms.

3.2.5 Inside out versus Outside in

Another decision to make when designing a visual tracking system is where to place the cameras. If the position of the camera is attached relative to the user, it is called inside-out tracking. This way of tracking uses any creature to find out its own position. The opposite would be outside-in tracking which is used e.g. by camera surveillance. Depending on their field of use both have their advantages and drawbacks.

The most obvious difference is the different range of use. Figure 5 shows different ranges of use. Inside-out tracking supports a circular range around the reference point in which the user is only allowed to look in the direction of the reference object. Outside-in tracking only supports a pyramid-shaped range of use with full freedom of rotation for the user. Of course, these limitations can be avoided with multiple camera systems.

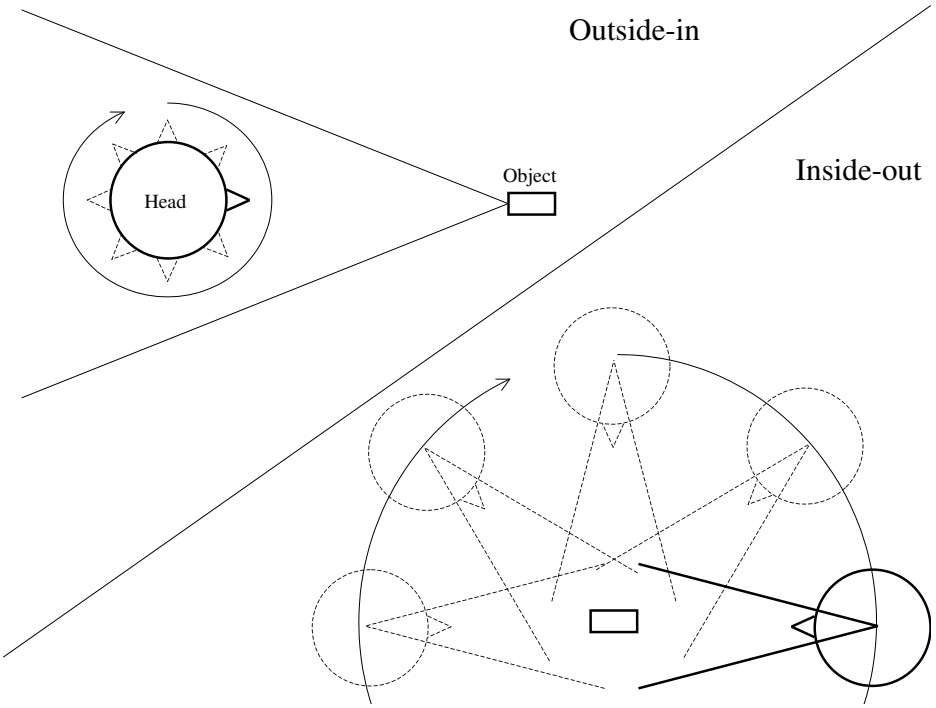

Figure 5: The range of use differs from the reference of the camera. In outside-in tracking the user may turn 360° and move in a pyramid shaped area. In inside-out tracking the user can move in a circular range around the reference object, but is only allowed to turn in the range of camera view.

Another reason for deciding whether to track relative from the user or not is how easy it is to attach a camera to the user or the environment. If a very large area is supposed to be tracked it will be easier and cheaper to use inside-out tracking if cameras can be attached to everyone using the system. On the other hand, if many users are supposed to be tracked at the same time or cameras cannot easily be attached to a user outside-in tracking should be the method of choice.

There is no difference in accuracy of two tracking techniques in principle if the same parameters have been chosen. In this context, there is an advantage in inside-out tracking if not only the head is tracked but a tool as well. The distance between a tool and the eye and the distance between the same tool and the tracking camera changes similarly in inside-out tracking as opposed to its counterpart. If a tool nears the head a higher accuracy is needed because it will be perceived larger and thus, smaller differences can be seen. On the other hand the accuracy of a visual tracker decreases with the distance to the

tracked object for exactly the same reason. In inside-out tracking this fits together very well. At closer distances between a tool and the head a higher accuracy is needed but a higher accuracy is provided by the visual tracker anyway. In outside-in tracking the accuracy of tracking does not necessarily change for the better if an object comes closer to the head.

4 Photogrammetric pose estimation

This section provides the theory behind registration via visual tracking. The fundamental problem of registration via computer vision is the task to discover the position and orientation of a camera in respect to the scene. This problem is also known as the problem of exterior camera calibration. In this chapter the transformations between the coordinate systems that are relevant in Augmented Reality are explained from the real object leading to the discretized data in the computer. It is explained in that way in order to make the models behind photogrammetric pose estimation easier to understand. Of course, the task in the system is the other way around: It is to deduce from the binary data to a model of the real object.

After showing the models of transformations the mathematical background is given that describes these transformations elegantly in projective geometry. The theory for exterior camera calibration is presented at the end of this section.

4.1 Coordinate systems

For registered augmentation is it necessary to model the relationship between points in space and pixels on the camera sensor as well as on the display. All of them are described in five different coordinate systems.

- **Object coordinate systems**. These three-dimensional coordinate systems describe relationships relative to a certain object. Models of objects are given in object coordinates.

- **World coordinate system**. This coordinate system is a three-dimensional coordinate system in respect to the scene. It is thought to be an absolute coordinate system because all of the objects are described relative to it. Its center and its axes can be chosen arbitrarily. These values can be chosen to make computation easier or to provide a demonstrative coordinate system.

- **Camera coordinate system**. This three-dimensional coordinate system is centered in the optical center of the camera. Its x and y axes are aligned with the the ones on the photo sensor. Its z axis points to the field of view. The camera coordinate system can be considered as a special object coordinate system.

- **Sensor coordinate system**. This two-dimensional coordinate system describes the image after projection on the sensor in metric measures.

- **Computer coordinate system**. This coordinate system is two-dimensional and pixel-oriented. The binary image data in the computer is given in this coordinate system. In contrast to the other coordinate systems it is discretized.

How transformations between these five coordinate system are realized is explained in the three following subsections. A fine introduction into the mathematical background is presented in [23].

4.2 Transformation between object, world and camera coordinate systems

Only transformations between world and object coordinate systems are explained in this subsection, because the camera coordinate system is a special kind of an object coordinate system. These transformations can be described by a rotation and a translation. A very common representation is an affine transformation with an orthonormal matrix for the rotation $\boldsymbol{R} \in \mathbb{R}^{3\times3}$ and a vector for translation $\boldsymbol{t} \in \mathbb{R}^3$. Any point ${}^{w}\boldsymbol{p}$ in world coordinates can be described as a point ${}^{obj}\boldsymbol{p}$ in object coordinates by

$$ {}^{obj}\boldsymbol{p} = \boldsymbol{R}\,{}^{w}\boldsymbol{p} + \boldsymbol{t} \qquad with\ \boldsymbol{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix},\ \boldsymbol{t} = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \tag{1} $$

If $\boldsymbol{R}$ and $\boldsymbol{t}$ represent the transformation from world coordinates to camera coordinates they describe the position of the camera in space. Hence they are also called *extrinsic camera parameters*.

4.2.1 Representation of rotation in space

The matrix notation offers efficient and numerically stable calculation on a computer. Unfortunately, this notation is not intuitive to read for humans. As there are only three degrees of freedom for rotations in space the representation of a 3×3 matrix obviously holds a lot of redundant information.

Three Euler angles

Any rotation can be represented by three subsequent rotations around the axes. The angles around the coordinate axes ϕ, ψ and θ are called Euler angles [36]. In matrix notation it is

$$ \boldsymbol{R} = \boldsymbol{R}_z(\theta)\boldsymbol{R}_y(\psi)\boldsymbol{R}_x(\phi) \tag{2} $$

and with rotation matrices in full length it is

$$\boldsymbol{R} = \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\psi & 0 & -\sin\psi \\ 0 & 1 & 0 \\ \sin\psi & 0 & \cos\psi \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} = \tag{3}$$

$$\begin{pmatrix} \cos\psi\cos\theta & \sin\phi\sin\psi + \cos\phi\sin\theta & -\cos\phi\sin\psi\cos\theta + \\ -\cos\psi\sin\theta & -\sin\phi\sin\psi\sin\theta + \cos\phi\cos\theta & \cos\phi\sin\psi\sin\theta + \sin\phi\cos\theta \\ \sin\psi & -\sin\phi\cos\psi & \cos\phi\cos\psi \end{pmatrix} \tag{4}$$

The three angles can be extracted from the rotation matrix via

$$\begin{aligned} \sin\psi &= r_{31} \\ \tan\phi &= -r_{32}/r_{33} \\ \tan\theta &= -r_{21}/r_{11} \end{aligned} \tag{5}$$

This representation has only three degrees of freedom but it contains singularities and evaluating trigonometric functions is time consuming and numerically sensitive.

One rotation around unit vector

Another common representation is a rotation around a normalized 3D vector $\boldsymbol{u}$ that is named unit vector. A vector $\boldsymbol{x}$ is rotated around the vector $\boldsymbol{u}$ with the angle ω by the formula:

$$\boldsymbol{x}_{rotated} = \boldsymbol{x} + (\boldsymbol{u} \times \boldsymbol{u} \times \boldsymbol{x}) + (\boldsymbol{u} \times \boldsymbol{x} \sin\omega) + (-\boldsymbol{u} \times \boldsymbol{u} \times \boldsymbol{x}) \cos\omega \tag{6}$$

$\boldsymbol{x} + \boldsymbol{u} \times \boldsymbol{u} \times \boldsymbol{x}$ is the orthogonal projection of $\boldsymbol{x}$ on $\boldsymbol{u}$ while the other terms perform the rotation. Replacing the cross products by a multiplication with the matrix

$$\boldsymbol{V} = \begin{pmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{pmatrix} \tag{7}$$

that computes the cross product with vector $\boldsymbol{u}$, we obtain the so called Rodrigues' Rotation Formulas [82]:

$$\boldsymbol{R}_u(\omega) = \boldsymbol{I} + \boldsymbol{V}\sin\omega + \boldsymbol{V}^2(1 - \cos\omega) = e^{\boldsymbol{V}\omega} \tag{8}$$

with $\boldsymbol{I}$ as the identity matrix and e as Euler's number. In the evaluated form it is:

$$\boldsymbol{R} =$$

$$\begin{pmatrix} \cos\omega + u_x^2(1-\cos\omega) & u_x u_y(1-\cos\omega) - u_z\sin\omega & u_y\sin\omega + u_x u_y(1-\cos\omega) \\ u_z\sin\omega + u_x u_y(1-\cos\omega) & \cos\omega + u_y^2(1-\cos\omega) & -u_x\sin\omega + u_y u_z(1-\cos\omega) \\ -u_y\sin\omega + u_x u_z(1-\cos\omega) & u_x\sin\omega + u_y u_z(1-\cos\omega) & \cos\omega + u_z^2(1-\cos\omega) \end{pmatrix} \tag{9}$$

Especially in computer graphics this representation is used because it does not possess the drawbacks of the Euler angle representation. To obtain $\boldsymbol{u}$ we can make use of the information that any point on the rotation axis will be projected onto itself or in a formula notation $\boldsymbol{R}_u s\boldsymbol{u} = s\boldsymbol{u}$ where s denotes any scalar value. This is a classical eigenvector problem, so $\boldsymbol{u}$ can be obtained by an eigenvalue decomposition.

In order to find out the angle ω we only need a special property of the trace of a matrix. Since $trace(\boldsymbol{BAB}^{-1}) = trace(\boldsymbol{A})$ is true for all square matrices $\boldsymbol{A}$ and $\boldsymbol{B}$,we can obtain ω by decomposing the rotation matrix into an unimportant rotation that is undone afterwards and a rotation around the z-axis:

$$\boldsymbol{R} = \boldsymbol{B}\begin{pmatrix} \cos\omega & \sin\omega & 0 \\ -\sin\omega & \cos\omega & 0 \\ 0 & 0 & 1 \end{pmatrix}\boldsymbol{B}^{-1} \tag{10}$$

From this we can obtain

$$trace(\boldsymbol{R}) = \cos\omega + \cos\omega + 1$$

$$\Leftrightarrow \cos\omega = \frac{trace(\boldsymbol{R}) - 1}{2} \tag{11}$$

Quaternions

Last but not least there is a notation that describes arbitrary rotations using normalized quaternions [3]. This notation has four degrees of freedom as opposed to 9 degrees in the matrix notation. The fact that its evaluation has less operations is losing importance since special hardware support of matrix rotations became common in graphics hardware in the last decade. Although this notation is not intuitive, it is used for theory of projection, because it yields some elegant proofs.

4.3 Transformation between camera and sensor coordinates

The transformation between camera and sensor coordinates has to model a 3D-2D projection that happens in the optics. The pinhole camera model is

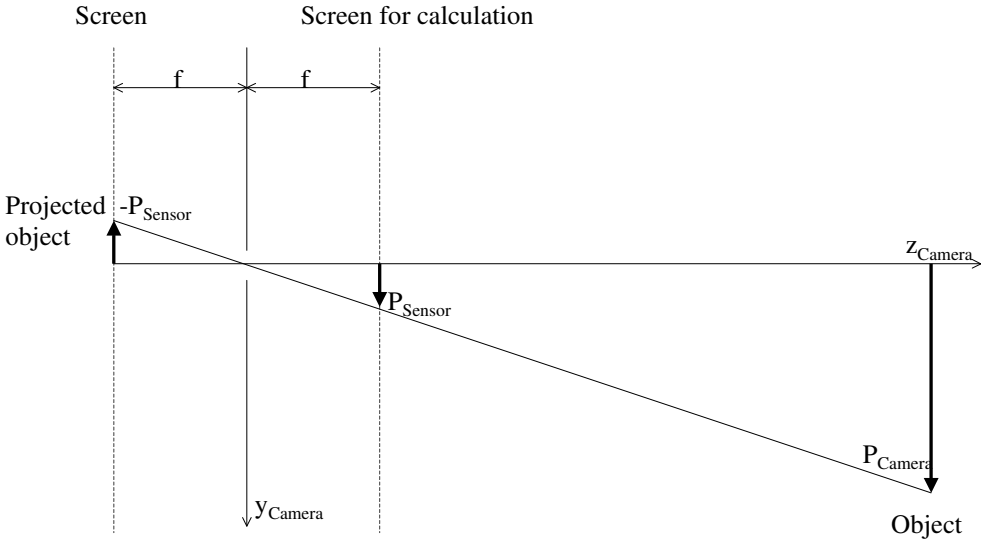

Figure 6: Illustrates principle of a pinhole projection. For easier calculation the screen is imagined to be mirrored to the other side of the pinhole.

the common camera model because it is efficient in computation while providing accurate results. This model is extended by modeling lens distortion in a subsequent step.

4.3.1 Camera model

In contrast to modeling projection via thin lenses which is common in optical physics, computer scientists prefer using the pin hole camera model in computer vision. It assumes that the object is projected through a pinhole or a lens with the correct focus. This simple geometric model can be handled easily with intercept theorems. The principle of projecting each point via lines through an optical center is called perspective projection. This is depicted in figure 6. Applying intersect theorems we obtain

$$\frac{P_{Sensor}y}{f} = \frac{P_{Camera}y}{P_{Camera}z} \tag{12}$$

For the x-Axis a similar equation can be found by replacing y with x. That leads directly to the formula for the pinhole projection:

$$P_{undistortedSensor} = \frac{f}{P_{Camera}z} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} P_{Camera} \tag{13}$$

The perspective projection models an ideal projection but in fact there are distortions caused by lenses. Therefore the point in this formula has the subscript 'undistorted'.

4.3.2 Lens distortion

There are different kinds of distortions with different impacts on the image. Especially cheap camera lenses suffer from radial distortion. If optics are made of spherical surfaces a geometric distortion occurs in radial direction. A point is optically projected to a point with a different distance to the image center than determined by perspective projection. If distances are distorted to the longer it is called barrel distortion. Otherwise it is called pin-cushion distortion. The function of distortion can be modeled as a power series [21]:

$$\delta P = P_d(\kappa_1 r^2 + \kappa_2 r^4 + \ldots) \tag{14}$$

with

$$r^2 = (P_{distortedSensor} - C)^2$$

This series can already be retained at the second term because higher terms do not add accuracy in prediction but increase numerical instability [7]. Hence, modeling radial undistortion results in the formula

$$P_{undistortedSensor} = P_{distortedSensor} \left(1 + \sum_{i=1}^{2} \kappa_i r^{2i}\right) \tag{15}$$

with r^2 as above. Note that it is on purpose that κ is defined in a way that evaluating the polynomial undistorts a point. This task is done more frequently in the case of photogrammetric pose estimation than its opposite. The inverse task, distorting, takes much more computation time if κ is defined in that way.

Another kind of distortion is tangential distortion. According to Horn [21] it can be modeled as

$$\delta P = P_d \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} (\epsilon_1 r^2 + \epsilon_2 r^4 + \ldots) \tag{16}$$

There are also other kinds of distortion in literature [4] but they are much less significant than radial distortion. For this reason, RAMP[10] takes into account only radial distortion.

4.4 Transformation from sensor to image coordinates

The image on the sensor is still a continuous image in metric measures. In this last step the sensor has to be modeled. By convention the center of the sensor coordinates is is the optical center. This is not necessarily the center of the sensor chip.

$$P_{image} = \begin{pmatrix} s_x/d_x & 0 \\ 0 & 1/d_y \end{pmatrix} P_{distortedSensor} + C \tag{17}$$

$C = \binom{c_x}{c_y}$ denotes the center of the image which is not for sure in the center of the image.
d_y denotes the distance between adjacent sensor elements in y- direction.
$d_x = d'_x \frac{N_{cx}}{N_{fx}}$
d'_x denotes the distance between adjacent sensor elements in x- direction.
N_{cx} denotes the number of sensor elements in a line
N_{fx} denotes the number of pixels in a line
s_x is the image scale factor
Except for s_x all of the parameters are self explaining. As a complicating factor in typical CCD or CMOS cameras, sampling in horizontal direction is not controlled by spacing of sensor cells in contrast to vertical direction [21]. The initial discrete horizontal signals are lowpass-filtered for a smooth video signal. Therefore the staircase signal is not preserved. This signal is digitized by the framegrabber[5]. The horizontal spacing between pixels does not generally correspond to the spacing between cells in the sensor.

4.5 Camera calibration

The procedure to find out all parameters to fill all of the matrices for a transformation from world coordinates to image coordinates is called camera calibration. It can be subdivided into two procedures dependent on the nature of the parameters:

4.5.1 Interior camera calibration

Interior camera calibration is about finding intrinsic camera parameters which remain static once the camera is assembled. These parameters only have to be estimated once for each camera. Hence, a time consuming algorithm may be used for estimating them. In the model of this chapter it is $f, C, s_x, \kappa_1, \kappa_2$ as unknown parameters to be estimated and $N_{cx}, N_{fx}, d'_x, d_y$ to be copied from the device specification. Note that as an exception f need not be static if the camera has a variable zoom objective.

4.5.2 Exterior camera calibration

Exterior camera calibration is about finding extrinsic camera parameters. These describe the position of the camera which can be divided into translation and rotation. Generally the position changes in an Augmented Reality system and it has to be calculated continuously. Therefore, a fast algorithm is needed for its calculation.

4.6 Projective geometry

Euclidean geometry is well suited to describe our world and its rigid objects well. Lengths and angles between objects are preserved if an euclidean transformation, i.e. translation or rotation, is applied. Also parallel lines remain parallel after transformation. Because it describes our world of rigid objects so well one might get the impression that Euclidean geometry is the only one of its kind. Optical phenomena cannot be described with it and they must be realized in a higher kind of geometry. Projective geometry is a geometry that describes perspective projections that occur in physics e.g. in pinhole cameras. Euclidean geometry is a subset of it. Similarity and affine geometry are two other kinds of geometry in between. The relationship between them is Euclidean $\subset$ Similarity $\subset$ Affine $\subset$ Projective. The more types of transformation can be performed the fewer invariants remain in that geometry. For an overview of possible transformation and invariants in different kinds of geometries see table 2. Information regarding projective geometry and much more about it can be found in [3], [82], [23] and in [88].

4.6.1 Homogeneous coordinates

Projective geometry can be described in homogeneous coordinates. To convert an Euclidean point to homogeneous coordinates a 1 can be added to the end of the the vector. For example, the point $\mathbf{a} = (x, y)$ in the Euclidean plane can be described as $\mathbf{a}' = (x, y, 1)$ in the projective plane, or the point $\mathbf{b} = (x, y, z)$ in Euclidean space is $\mathbf{b}' = (x, y, z, 1)$ in perspective space. Generally, the scaling in homogeneous coordinates is unimportant. Any point $(\mathbf{p}, w)^T$ can be given as $(\alpha\mathbf{p}, \alpha w)^T$, $\alpha \neq 0$. Therefore they are called *homogeneous* coordinates. Because of $\alpha \neq 0$ the point (0,0,0,0) does not belong to the perspective space. Since the scaling is important in Euclidean coordinates, homogeneous ones must be scaled back to $w = 1$ for conversion!

With homogeneous coordinates a rotation and a translation can be ex-

Transformation	Euclidean	Similarity	Affine	Projective
Translation	✓	✓	✓	✓
Rotation	✓	✓	✓	✓
Scaling		✓	✓	✓
Shearing			✓	✓
Projection				✓
Invariants				
Length	✓			
Angle	✓	✓		
Ratio of lengths	✓	✓		
Parallelism	✓	✓	✓	
Centroid of an ellipse	✓	✓	✓	
Incidence	✓	✓	✓	✓
Cross ratio (see 4.6.2)	✓	✓	✓	✓

Table 2: Overview of geometries versus their transformations and invariants.

pressed as a single matrix multiplication.

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix} \tag{18}$$

The inverse of it is simply

$$\mathbf{T}^{-1} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} \mathbf{R}^T & -\mathbf{R}^T\mathbf{t} \\ 0 & 1 \end{pmatrix} \tag{19}$$

The pinhole projection can described as

$$\begin{pmatrix} x \\ y \\ w \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} P_{Camera} \tag{20}$$

However, the resulting vector must be scaled back to $w = 1$. The whole projection (without modeling distortion) can be expressed as elegantly as

$$\mathbf{p}_{image} = \mathbf{T}_{internal}\mathbf{T}_{perspective}\mathbf{T}_{external}\mathbf{p}_{world} \tag{21}$$

There are also other models like the ray space, unit sphere or augmented affine plane, but for computer vision the most common way of describing coordinates in projective geometry is homogeneous coordinates.

4.6.2 Cross ratio

The cross ratio is a measurement that is defined for four collinear points.

$$CR(A,B;C,D) = \frac{\overline{AC}\cdot\overline{BD}}{\overline{AD}\cdot\overline{BC}} \tag{22}$$

where $\overline{XY}$ denotes the signed distance between the points X and$Y$. Signed distance means $\overline{XY} = -\overline{YX}$. Although the cross ratio is invariant once the order of the points has been chosen, its value is different depending on their order. Six distinct values can be obtained from the 24 combinations which are related by the set

$$\left\{\tau, \frac{1}{\tau}, 1-\tau, \frac{1}{1-\tau}, \frac{\tau-1}{\tau}, \frac{\tau}{\tau-1}\right\} \tag{23}$$

Due to these six different possibilities the cross ratio can be defined differently than in equation 22 by permutation of the points.

Cross ratio can also be obtained from five coplanar points. (See figure 7) The cross ratio of the intersections of the four lines with another line is constant if the line crosses all of the four lines and does not incide with F. This feature does not really surprise taking the fact that the cross ratio remains the same after perspective projection. Yet another way of calculating the same value is

$$Cr(A,B,C,D,F) = \frac{\sin(\measuredangle AFC)\sin(\measuredangle BFD)}{\sin(\measuredangle AFD)\sin(\measuredangle BFC)} \tag{24}$$

This can be obtained by applying the Sine- Theorem for angles in triangles with their opposite line.

$$\frac{\sin(\alpha)}{a} = \frac{\sin(\beta)}{b} = \frac{\sin(\gamma)}{c} \tag{25}$$

4.6.3 Conics (Ellipse, Hyperbola, Parabola)

Second order conic sections lose their distinction in projective geometry. This means that ellipses, hyperbola and parabola can be transformed in either of them. Collectively these forms are called conics. A conic can be expressed as a quadratic form given through the equation

$$\mathbf{p}^T\mathbf{C}\,\mathbf{p} = 0 \tag{26}$$

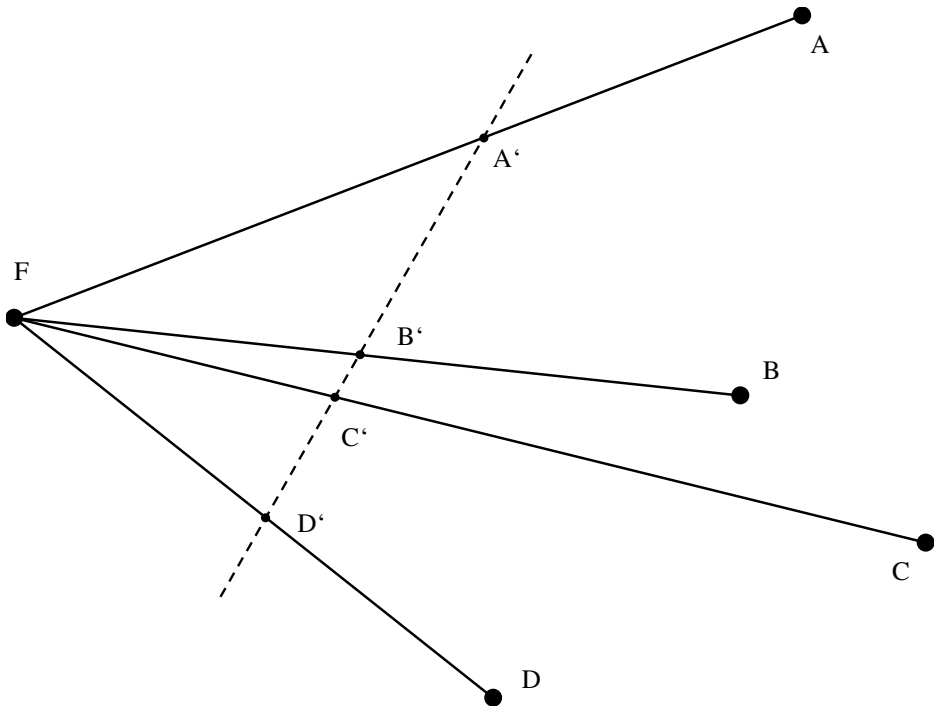

Figure 7: The cross ratio $Cr(A', B', C', D')$ is constant for any line that crosses the four others and that does not incides with F.

or

$$c_{11}x^2 + c_{22}y^2 + c_{33}w^2 + 2c_{12}xy + 2c_{13}xw + 2c_{23}yw = 0 \tag{27}$$

The matrix $\mathbf{C}$ is a symmetric matrix. A conic can be described by five of its points. Dividing by one of the redundant parameters, e.g. by c_{11} its parameters can be obtained by five equations provided by the points.

Ellipses can also be described by their centroid $\mathbf{M}$, the length of their semi-axes a, b, and an angle of rotation φ. This kind of description is popular in image processing because these measurements can be obtained directly from the image (see 5.3). The equation of an ellipse in euclidean geometry is

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \tag{28}$$

with $\theta = 0$ and $\varphi = 0$, but in general

$$\mathbf{p}^T\mathbf{R}^T \begin{pmatrix} 1/a^2 & 0 \\ 0 & 1/b^2 \end{pmatrix} \mathbf{R}\,\mathbf{p} + 1 = 0 \tag{29}$$

with

$$\mathbf{R} = \begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix}, \quad \mathbf{p} = \left(\begin{pmatrix} x \\ y \end{pmatrix} - \mathbf{M} \right)$$

Note the connection between equation 27 and 28 via equation 26 and 29.

4.7 The Perspective n-Point Problem

The exterior camera calibration can be calculated by equating a number of unknown points in the camera coordinate system with their corresponding points in the image. This mathematical problem is called the Perspective n-Point Problem (PnP). It has already been addresses as early as 1795 by Lagrange, solved for three points in 1841 (as stated in [2]), but it became popular in the late 1970s when computer technology offered sufficient computational power to use the solutions for photogrammetric purposes. The renaissance of the space resection problem, as it is also called, happened in in the field of cartography.

There are many different solutions with several advantages and drawbacks to them. In general we know that we need at least three points for exterior camera calibration, but they might be ambiguous [1]. There are two different approaches to the problem. There is an algebraic approach on one hand and an optimization on the other hand. There are different algebraic solutions depending on the number of points taken into account.

- **Three points**. Three points provide six different equations for the 3 degrees of freedom for each rotation and translation. These can be derived into a closed form expression of a bi-quadratic polynomial with one unknown. Therefore, three points provide a maximum of four solutions. Wolfe et al. [1] give a geometric insight into how many solutions have to be expected in which constellation. In that paper it is explained why it is important to care about all of the possible solutions if there is more than one. Multiple solutions occur not only in special cases, even though only the smallest group of cases holds four solutions. Reviewer of the journal article [2] compare three different kinds of closed form expressions with different numerical behaviors of robustness and accuracy.

- **Four and five points**. Fischler and Bolles [8] offer a non-ambiguous solution with coplanar points. They show that for four non-coplanar points two solutions may only exist as in the case for five points. They try to attack the problem by calculating the solutions for all combinations of three points within the set of four points and take the solution

all sets of solutions have in common with less success than the following: Another approach is to calculate only the solutions for three of the four points and verify the best solution with the remaining point. Other kinds of solutions are presented in [56], [76], and in [20] for five points .

- **Six points**. There is a very straightforward solution for P6P. For 6 points we obtain 12 linear equations. This is sufficient to determine the 9 coefficients of rotation and the 3 coefficients of translation of the homogeneous projection matrix.

The other solution is non-linear optimization of the external camera parameters. The parameters are changed in an iterative process until they fit the measurements. A very robust and efficient method is the Levenberg-Marquardt- Algorithm [6]. It is the commonly used algorithm for non-linear optimization problems.

To summarize, the algebraic solutions have the advantages of computational efficiency and they do not need starting values. On the other hand the optimization algorithms are better conditioned and consequently yield more accurate results. To take advantage of both approaches it is possible to combine them to a hybrid solution. First, employing an algebraic algorithm, we obtain a good starting value. This task is done in a short time. Since the starting value is already very close to the final result it can be computed much faster compared to optimization with a starting value gained from guessing. The results will be more robust against noise and more accurate than the mere algebraic solution. The most popular one is probably Tsai's approach [7]. That paper does not only offer a robust solution for 5 coplanar or 7 non- coplanar points. The described algorithm includes an interior camera calibration as well. Another combination of a rough algebraic approach and subsequent iteration is POSIT [19].

4.8 Identifying markers

In order to obtain a camera position by four points or more we must know their pendants in a model. Thus we need to measure the model beforehand but this is not a problem. A more difficult issue is ordering each point according to its counterpart in the model. Without any further information we would have to try each combination: $4! = 4 \cdot 3 \cdot 2 \cdot 1 = 24$. This brute force method can be done in a pretty short time, but it has two enormous disadvantages. How do we know which one is the right solution? Even if we could find out the right one by adding other points there is another huge drawback. For several reasons, like tracking different sets of markers

or adding numerical stability, we might need more than four points. We know that algorithms with a complexity like that of the faculty function have computational disadvantageous characteristics. To show how bad this complexity is even if we add only a few points there is an impressive number. The current system uses a maximum of 25 markers. The brute force method would take about $25!/24 = 646300418472124416000000$ times longer than checking each combination of four points, because a mere 21 markers have been added. So, combinatorics are against us. There are different ways of identification. These are the common means for identification

- **Adding information**. Adding barcodes or symbols to each of the fiducials[4] is an easy way of identification. This is a common approach used in different Augmented Reality systems. Various barcodes have developed for identification [40], [41], [43] to name a few. Another possibility is to use arbitrary symbols together with pattern recognition as in the popular ARToolkit [24]. Problems arise in all of these systems due to the size of the barcodes or symbols. They must cover a certain size in the image because they must be detected and evaluated robustly. This limits the number of possible fiducials to have in a single image, but in order to track multiple moving objects with a single camera many fiducials must be visible. Furthermore, for high accuracy results the redundant information of several fiducials can be merged. As many markers as possible should be available, for this technique.

- **Pulsating light**. If the fiducials emit light anyway it is possible to make the light blink. The frequency or the pattern of blinking can be used for identification. This is used e.g. in the commercial product Optotrak [87]. This way of identification allows for small fiducials but the production of such fiducials is more complicated and more expensive. Additionally, these fiducials are dependent on a source of power which is usually a battery.

- **Colors**. Fiducials may be identified by their color as in [53]. This is another easy way of identifying fiducials but it does not work with infrared images. Infrared vision might be desirable (see section 3.2.4).

- **Spatial constraints between fiducials**. This is the choice of RAMP[10]. Fiducials can be identified if they belong to a set of fiducials with fixed relative coordinates i.e. they belong to one and the same rigid body. Unfortunately, distances and angles are not generally preserved by perspective projection (see table 2). There are only a few properties that are preserved by perspective projection. The cross ratio

of lines and angels (see section 4.6.2) remains the same. Also the type of lines and ellipses and their intersections remain the same.

5 Accuracy considerations

As stated in section 1.2, accuracy is the main concern in at medical Augmented Reality. In section 3 it has been shown that visual tracking with fiducials yields the most accurate results. This section tells in detail how high accuracy can be maintained.

5.1 Choice of HMD

Accuracy is also a question of displaying technology. There are different types of displays used in AR [17] as HMDs[6], handheld displays or projectors. In medical Augmented Reality HMDs are mainly used because the user has both hands clear and projection onto the skin would suffer from distortion and bad contrast. Probably most important is the fact that HMDs can provide stereo vision which improves considerably handling of complicated tasks in space [81].

There are two kinds of HMDs. One technology projects the artificial images on a partial transparent mirror. The real environment can be perceived through the mirror. This kind of technology is called *optical see-through*. The other kind of HMDs have an opaque display. The real environment is recorded by an additional camera. Real and artificial images are merged in the computer. Therefore it is called *video see-through* technology. Figure 8 illustrate both kinds of HMDs. The difference in technology causes considerably different performances. A comparison is given in [16]. Depending on the field of use it can be both an advantage and a drawback.

- Real objects can be perceived in optical see-throughs without any latency. Therefore real images have no latency but the difference to virtual objects can be perceived. Real objects in video see-through HMDs suffer from latency but they can be synchronized with the virtual objects so that there is no time lag between virtual and real objects. As a rule of thumb according to [17] the lag is responsible for one additional millimeter error per millisecond for short range tasks.

- The real environment cannot be switched off with optical see-through HMDs. That means in case of a malfunction of the system the real world can be still perceived. If a video see-through stops working the real environment cannot be seen at all. Because the real objects cannot be darkened in any way with optical see-though technology, several problems arise. First, it is impossible to remove objects from the scene as they always shine through the artificial cover. Second, virtual objects have always a ghostly and unreal nature because they cannot conceal

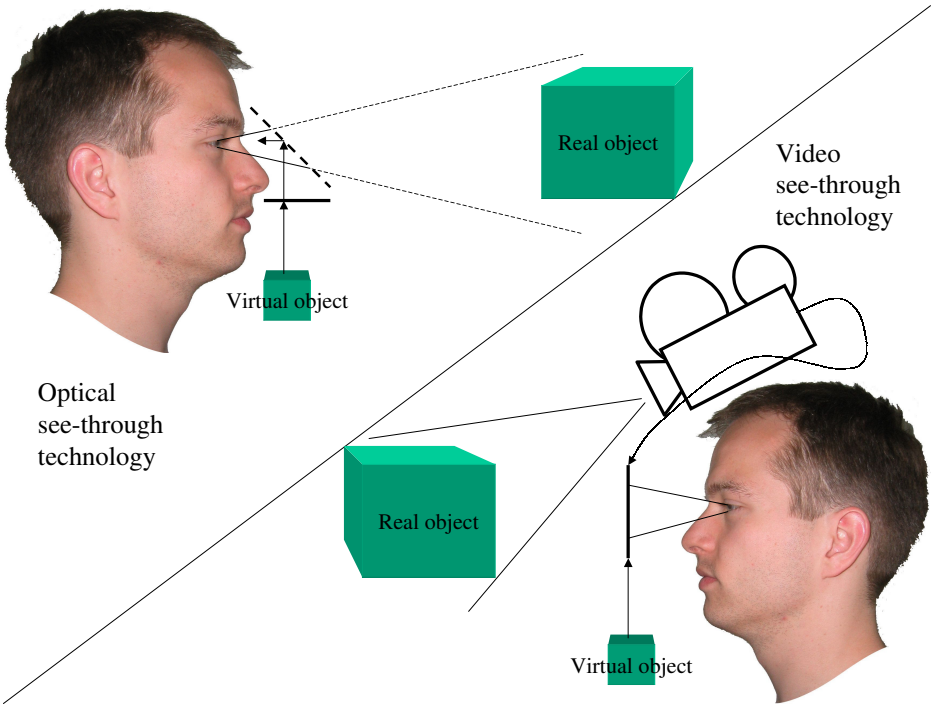

Figure 8: Both technologies aim at combining virtual and real images. While video see-through HMDs combine the images electronically the other HMDs do this optically.

real objects. Third, the depth information is misinterpreted by our brain because the depth cues through are missing. Occlusion is one of the most powerful [39] depth cues.

- Another difference is the quality of real objects. In optical see-though HMDs real objects have brighter colors, a higher resolution, the correct accommodation distance, etc. Therefore, real objects remain with the best possible quality while virtual objects suffer from the drawbacks of the display. What seems to be a advantage of the optical solution can turn out to be drawback as well. The difference in quality makes virtual object look even more artificial. Both, real and virtual objects are shown on the same display using video-see-through technology. This approach provides the same image quality. Therefore relative depth cues between virtual and real object are more consistent leading to a more exact perception of virtual objects [37].

- Since the semi-transparent mirror of an optical see-through HMD does

not have a fixed position relative to the eye it must be calibrated each time it has been moved on the head. This can be done either by asking the user for a calibration procedure as in [27] which had to be repeated each time the HMD had been moved slightly. If we wanted to maintain a certain accuracy for dependability we would have to track the eye-to-HMD position all the time causing extra costs and adding another source of possible errors.

As always in this comparison, video see-through technology does not suffer from differences between virtual and real objects but differences between objects on the display and objects seen without a display. As can be seen in figure 8, for video see-through HMDs the point of view is moved to the camera. Because both, real and virtual objects, are shifted to the camera there is no problem of eye-to-HMD calibration. Then again, it might be disturbing to see the environment including one's own hands from a different point of view. In addition to the transposition a distortion of depth can be perceived if the distance of the cameras is different from the displays. The depth cue of convergence is misleading in this case.

- As a last difference just for video see-through technology, real and virtual images are digitally available in separate form as well as combined. This enables the system to record for both archivation or transferring it somewhere else for teleoperation. Furthermore, it is easy to provide a digital zoom.

For medical Augmented Reality video see-through technology has more advantages than its optical pendant. The alignment error of the display, that is the relative error between real and virtual objects, is known in advance to be zero. There is no alignment error from eye-to-HMD calibration and there is no alignment error from a time lag. The feature of optical see-through HMDs that alignment error is simply unknown is in flagrant contradiction to the requirement of dependable accuracy (see section 1.2). Also the fact that the alignment of real and virtual may be geometrically correct but is misperceived with optical see-through HMDs runs against the aim of dependability.

5.2 Feature extraction

Generally in computer vision it is an important question what to look for. The environment of medical Augmented Reality allows for the use of fiducials. For accuracy and robustness their use is inevitable. Fiducials can be tailored to provide highest accuracy while offering easy processing which is important

for the real time constraint. For highest accuracy a subpixel-accurate feature must be chosen.

There are many different ways of subpixel accurate feature extraction in images instances are interpolation between pixels, pattern recognition, template fitting and obtaining moments of areas. The latter one is very fast in computation, very robust and easy to implement. As moments are the main feature of RAMP's[10] fiducials, it will be explained in detail.

5.3 Moments of an area

Given any image stored in array $\mathbf{B}$ the moments of an area can be computed as follows.

- **Zeroth order moment: Size**.

$$A = \sum\sum \mathbf{B}_{i,j}$$

- **First order moment: Position or Centroid** .

$$\mathbf{P} = \frac{1}{A}\begin{pmatrix} \sum\sum i\mathbf{B}_{i,j} \\ \sum\sum j\mathbf{B}_{i,j} \end{pmatrix}$$

- **Second order moment: Orientation and Perimeter (if ellipse)**.

$$\mathbf{C} = \frac{1}{A}\begin{pmatrix} \sum\sum(i-\mathbf{P}_x)^2\mathbf{B}_{i,j} & \sum\sum(i-\mathbf{P}_x)(j-\mathbf{P}_y)\mathbf{B}_{i,j} \\ \sum\sum(i-\mathbf{P}_x)(j-\mathbf{P}_y)\mathbf{B}_{i,j} & \sum\sum(j-P_y)^2\mathbf{B}_{i,j} \end{pmatrix}$$

  Orientation:

$$\tan 2\varphi = \frac{(c_{12}+c_{21})}{(c_{11}-c_{22})}$$

  Perimeter of an ellipse (for theory see Appendix section 9.1):

$$a = 2\sqrt{\frac{d_{11}}{A}}$$

$$b = 2\sqrt{\frac{d_{22}}{A}}$$

  with $\mathbf{C} = \mathbf{VDV}^{-1}$ (Eigenvector decomposition). The matrix $\mathbf{V}$ describes a rotation around the center with the angle φ.

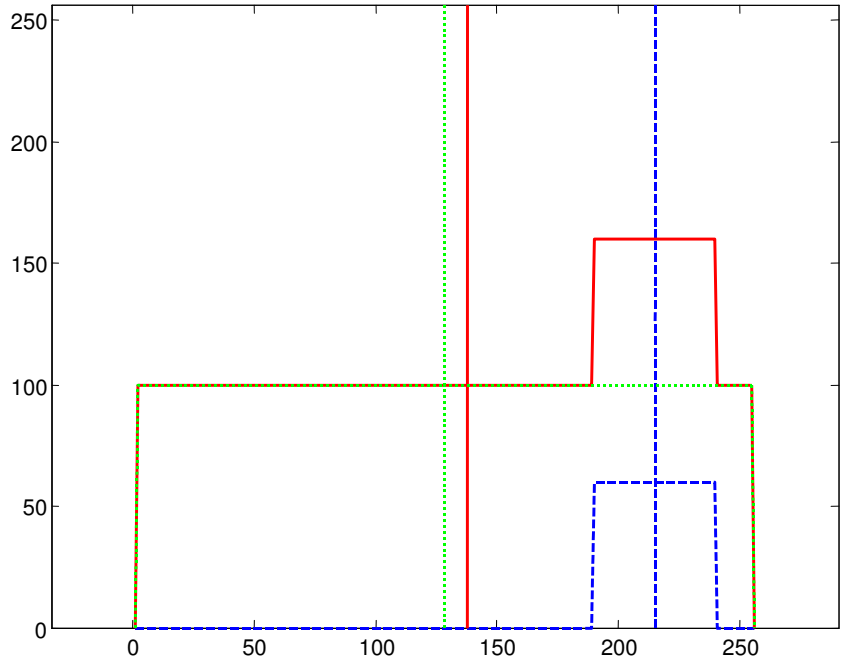

Figure 9: Bias affects measurement of center at gray scale images. The dashed blue line shows the function and its center of gravity. The dotted green lines shows the function of bias that is constant for illustration purposes. The solid red lines show the combination of both the function and its center.

These formulas are just correct for binary images and for gray scale images if the background of the image is zero. The average gray scale value of the background is called bias. For moment calculation of gray scale areas the bias must be subtracted from each pixel ($B_{i,j}$ in the formulas). Figure 9 visualizes the problem at the first order moment. The bias contributes an error that drags the first moment to the center of the image which is not necessarily the center of the region. Grayscale regions yield more accuracy than simple binary region moment extraction.

This looks pretty good, but the centroid of the ellipse is not necessarily the projected center of the circle. Figure 10 shows the difference between these two points. To find out if this difference is relevant for computation we have to calculate.

$$err = I_1 - (I_1 + I_2)/2 \tag{30}$$

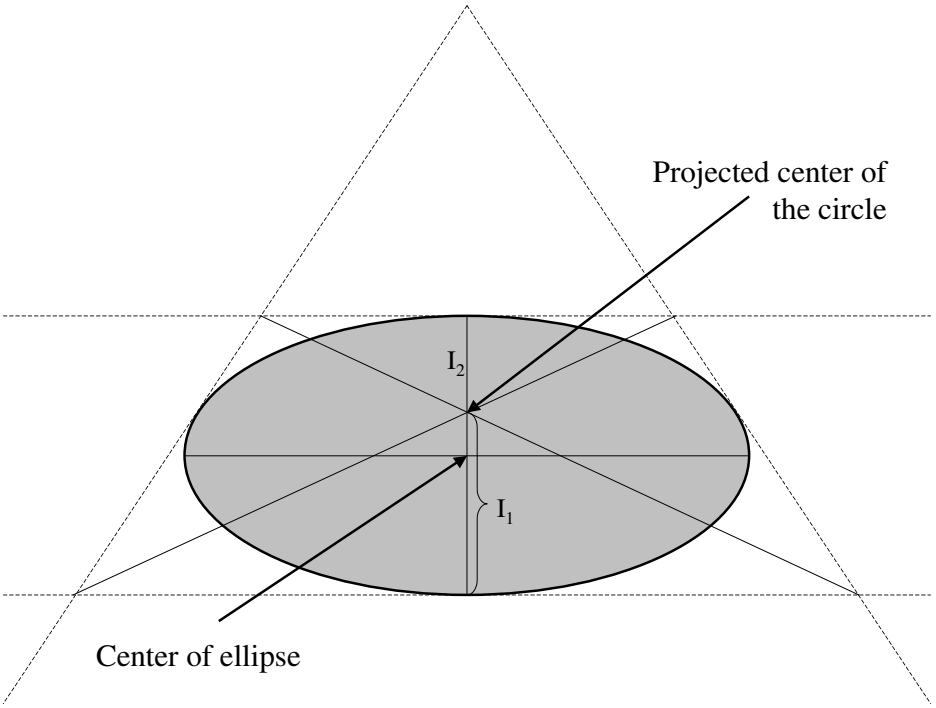

Figure 10: Perspective projection of an ellipse does not generally preserve its center.

I_1 and I_2 can be calculated as in section 9.2.3. Figure 52 illustrates the geometry that is the basis for the following function.

$$err = \frac{r^2 \sin\alpha \cos\alpha}{d^2 - r^2 \cos^2\alpha} \tag{31}$$

Figure 11 shows a plot of that function with $d = [350; 1000]$mm, $\alpha = [0°; 90°]$ and $r = 9.5$mm which are the boundaries of RAMP[10]. The highest value of this plot is 0.00035 mm. This is far less than the accuracy of the system. For this reason the centroid of the ellipse can be taken as an approximation of the projected center of the circle in RAMP's environment without losing accuracy.

5.4 Sources of errors

In order to get a satisfactory pose estimation we need measurements in the image to be as accurate as possible. The estimates can only be as accurate as the measurements they are based on. A major problem in image processing

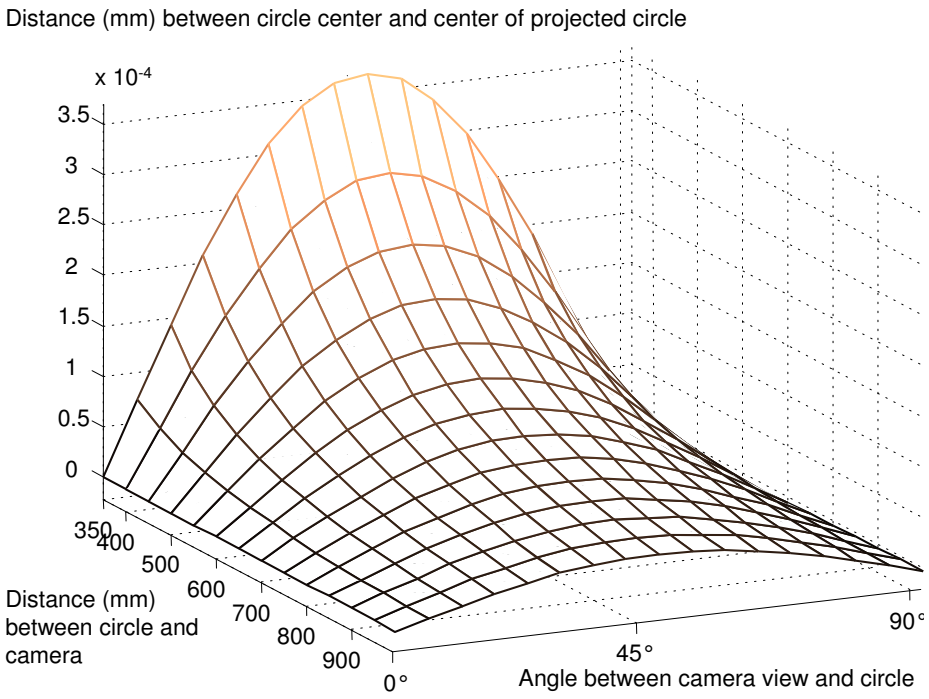

Figure 11: Plot of equation 31 reveals that the highest value of the function in the boundaries of RAMP's range of use is far below a thousandth of a millimeter

is avoiding measurement errors. One major source of errors are sensor errors; especially photo sensors provide noisy data ([22], p.38).

Another source of errors are the lighting conditions. This challenging problem has been treated in section 3.2.4. A completely different kind of error source is of numerical nature. Some ways of calculating estimates yield higher errors than necessary because ill-conditioned algorithms are used. A fine example are algebraic solutions of the perspective three-point problem. In [2] it is examined that different kinds of approaches to one and the same problem lead to different numerical behavior. Last but not least, modelling is another source of error. The model of the physical processes can only be an approximation of reality. Thus the implicit and explicit models of a system must be checked if their approximation is accurate enough for the desired overall accuracy. This is exactly what has been demonstrated at the end of section 5.3.

5.5 Increasing accuracy of estimates

The more redundant information (with an independent source of error) we can find and add together, the smaller is the error in the measurement and probably also in the final results. In other words, we can increase the accuracy of measurement by taking the position into account of more than one pixel. The more pixels the better. In principle, features of several pixels size yield more accurate information than only one pixel. Additionally, it is advisable to use the gray scale information of each pixel and not only its position. The problem about maximizing the amount of pixels used for one single feature is the finite amount of pixels in an image. If the features get too large there will be too little flexibility of movement left for the camera, because the points would exceed the boundaries of the image or would tend to occlude each other. So the number of pixels taken into account to calculate the coordinates of a single point in the image will always be a trade-off between flexibility and accuracy.

Pose estimation is done by transforming the information in the measurements of fiducials to transformation matrices containing translation T and rotation R. There are two ways of improving estimates without changing the quality of each measurement. The first one affects translation and rotation. Refinement of measurements can be achieved by taking into account more measurements than necessary to solve the equations and then optimize them. The idea is the same as above but it has different results. This one suggests more fiducials while the one above suggests bigger fiducials.

While the first techniques are not too difficult to discover, the second one is less apparent but still easy to understand. This technique affects just the quality of rotation but it may improve the augmentation considerably. After all, the rotations are calculated from discrete points. If these points are close together the ratio of measurement error to point-to-point distance increases and therefore the error of the estimated angle increases without any change in quality of each measurement. Figure 12 visualizes the problem and its effects. In order to keep error as small as possible

- augmented points should be as close as possible to the fiducials
- the space between fiducials should be as large as possible

The impact of space covered and the number of fiducials for accuracy has been examined in [63].

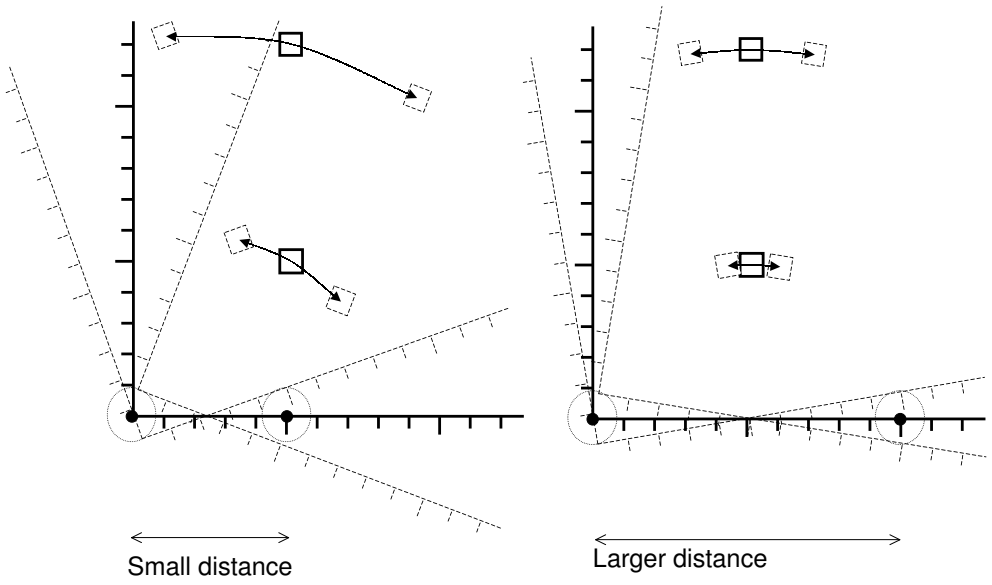

Figure 12: Visualizes the effect of numerical leverage at coordinate systems. Round dots represent measurements that hold the coordinate systems. Dotted circles visualize a certain confidence area of the measurements of these points. Dotted coordinate systems show two examples of coordinate systems that could have been generated with other measurements from the confidence area. Boxes are projected according to each coordinate system on (5,5) and (5,12). They serve as examples. We note first that the confidence areas of the the boxes decrease with a higher distance between the measurements. Secondly, the error increases at a higher distance between the point and the coordinate axes as can be seen from the box that is further away.

5.6 Jitter

Jitter is the unpleasant effect of noise in tracking which is perceived as a vibration of augmented objects that are not supposed to be moving at all. This effect is most obvious if the head is not moving. A common approach to reduce jitter is filtering the results over time (e.g. in [65] or in [64]). This is usually done with a Kalman filter [48]. Filtering the results over time contradicts the paradigm of dependable accuracy of medical Augmented Reality. First of all, jitter reminds the user how accurate the system is. Filtered data represents an accuracy that does not exist. More important is the following argument. Filtering makes the error dependent on the preceding image sequence. This can increase the error. Therefore the maximum error possible is increased by filtering. Filtering does not improve but reduces the accuracy as far as the maximum error is concerned. The only way of reducing jitter in medical Augmented Reality is recording more accurate measurements. The limits of perception of jitter are illustrated in the next paragraph.

5.7 Aimed accuracy

There is no such as 'too accurate' but it is an interesting question what accuracy can be perceived. This is on the one hand the accuracy that is aimed but on the other hand it is the accuracy that need not be improved any more for AR systems. The highest resolution of the eye is 1/60 degree (see section 2.1). Therefore, when augmenting an object at double this accuracy, no error at all can be perceived. The perceivable error in translation is proportional to the distance of an object to the eye. At a $50cm$ distance to the eye which would be a common distance to the hands, the visual accuracy is as high as $0.15mm$. Therefore no one would notice jitter of $0.075mm$ even if the the display's resolution was high enough. A current high end display may have a vertical resolution of 1024 pixel and field of view (FoV) of 35 degrees which makes an acuity of 1/29.26 degree.

The aimed accuracy has to be specified at one point in the system. The iterative algorithms for camera calibration need tight stopping criterias to fulfill the real time constraint of the AR system. The three values of each, translation and rotation are optimized to the given data. Unfortunately, the optimization algorithm treats the millimeters gained from the translation just as it treats the degrees gained from the rotation, although a difference of one degree does not necessarily translate into a difference of one millimeter.

Hence, these values must be weighted according to this formula

$$1\,degree \equiv d \cdot \lim_{h \to 0} \left(\frac{\tan h}{h} \right) \approx d \cdot\ 0.01745 \tag{32}$$

where d denotes the distance from the camera to the object. The formula is straightforward except for the limit function. Of course, an angle cannot be expressed linearly through a distance. The approximation assumes that the angles are near zero.

Let us have a look at examples for systems for close range work. At a distance of $50cm$ for instance, degrees and millimeters must be weighted $1 : 8.72$ or $1 : 17.45$ at $1m$. This value has to be set for a camera calibration by non-linear optimization. Fortunately, a starting value has to be set for the optimization anyway, so we roughly know the distance between camera and marker. By this means, the linear weighting factor to convert angles to a distance can be set dynamically.

5.8 Real time constraint

Real time as used in [16] for the definition of Augmented Reality (see section 1.1) is a fuzzy term. It is used but no specific definition is given. One could paraphrase it by 'on the fly' which is not precise either. Real time constraints of an AR system should not be confused with the ones of an operating system. Interaction in real time addresses two parameters of an AR system. First, there is the lag of the presented data; second there is the update rate. If the user cannot perceive any distortion in time during interaction with the system, it is perceived temporally as a real interaction, so it can be called real time interaction. In the case of visual augmention the criteria are very tight. The update rate for new images must be between $20Hz$ and $30Hz$ to maintain the the impression of apparent motion. Apparent motion, also named Phi-phenomenon, is based on the ability of the brain to fuse two points presented subsequently as a single moving point. This is dependent on the personal flicker rate. The lag that cannot be perceived must be far below $10ms$ as stated in [17]. Especially the lag makes the real time constraint too tight. Hence, we also call a system real time if it does not respond perfectly without temporal distortion. It is enough that it gives the impression of an almost real feedback.

On order to maintain a certain update rate all of the calculations must be done in a certain frame of time. Time consumption of some of the algorithms depends on the input but just a fixed amount of time is available. Therefore just the worst case time consumption has to be examined. If the time frame

is not exceeded in the worst case the update rate can be maintained, but a user will not benefit from a faster computation of average cases.

6 RAMP

6.1 Outline of RAMP

RAMP[10] is an Augmented Reality project of SCR[11] designed to support medical procedures. The name RAMP is an acronym for "Real Time Augmentation for Medical Procedures". Doctors are supported during many medical procedures by imaging data such as ultrasonography[12], CT[3] or MRI[9]. Imaging data can be very helpful because complex data can be provided in a way that is easy to understand. In clinical practice, images are shown in a crude way either on prints or displays far away from the patient (see figure 13).

RAMP is supposed to improve medical procedures in four different ways. It provides these features:

- **3D- Visualization**. Tomography imaging provides 3D imaging data but it is usually presented on normal displays that can just provide 2D images. By presenting 3D structures on an HMD[6] that supports stereo vision, depth information can be presented in a natural and intuitive way.

- **In-situ visualization**. Imaging data can especially support doctors performing minimal invasive surgery, because it is not possible to see through the skin. Images are usually presented on computer displays but a doctor's focus of attention is his hands operating. Therefore surgeons have to take a distracting glance away from the patient to the display (see figure 13). The solution is overlaying images in the doctor's field of view of the patient. This provides the ultimate closeness of images to the field of interest.

- **Image registration**. In most cases, the information in medical images is worthless without knowledge about where they have been taken. Image registration is the task of finding the relationship between the image coordinate system and the desired one. In this case registration is the answer to the following question: Where is the corresponding point in reality to each pixel on the display? Usually, the 3D registration is done in the doctor's brain, consuming energy he might want to spend on tasks computers cannot do. By overlaying the point in reality with the pixel of the image the registration is done for the doctor.

- **Tool registration**. Registered tools can be augmented for guidance. Tools like a biopsy needle can be shown even if they are concealed by tissue [61]. Further visual guidance can be given by showing the

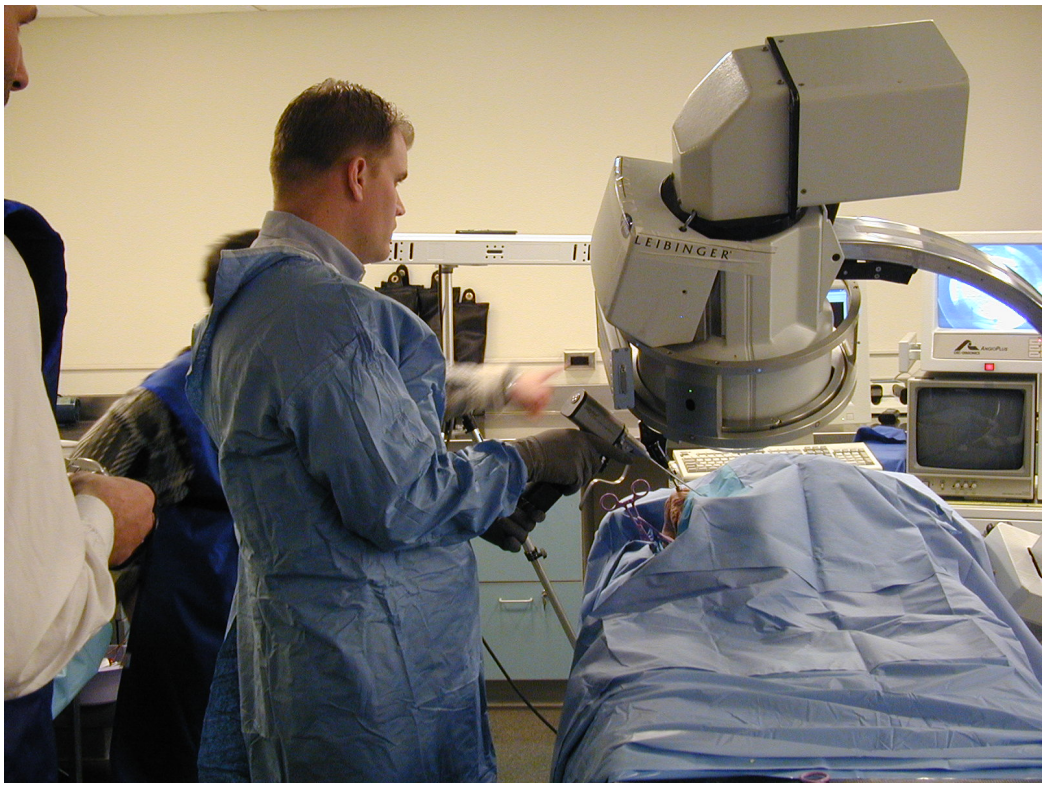

Figure 13: A setup for minimal invasive surgery. The doctor has to look away from the patient in order to see the display of the fluoroscope in the right hand corner

distance between the tool and any segmented area. This area can be the target area, e.g. a cancerous tissue or an area to avoid like blood vessels or nerves that should not be harmed.
Second, the position of registered tools can be recorded together with the camera views to understand the curing process after the surgery. By this means an invasive procedure can be documented easily.

When tackling these problems there are special constraints that are typical for RAMP[10]. The system must be compatible to a clinical environment. Registered tools must be disinfected without harming the tool and the system must be compatible to imaging machines (i.e. in a MRI scanner fiducials must be non-magnetic) as well.

Visualization	Video see-through HMD
Tracking	Visual, non-hybrid
	Inside-out
	Fiducial-based
	Infrared flash
Fiducials	Retro-reflective material
	Flat, circular shape
	Same appearance of each fiducial
Cameras	Left eye color, interlaced
	Right eye color, interlaced
	Tracking infrared, interlaced
Computation	Dual processor PC
	Graphics card with acceleration for 3D computations
Operating system	Windows 2000
Developing language	C/C++

Table 3: Short overview of RAMP

6.2 Description of the existing RAMP hardware

The aim of RAMP[10] hardware components is to meet cost constraints while delivering highest performance. That is why mainly off-the-shelf components are used. Table 3 gives a short overview of the RAMP system. While this section just describes the hardware, section 6.3 gives detailed information why RAMP has been composed in this way.

Visualization is provided by a video see-through HMD[6]. Therefore RAMP[10] needs two color cameras to copy the original field of view of the user to the HMD. The virtual objects are overlaid on these images. The third camera takes images in the near infrared spectrum for optical tracking. The infrared spectrum is used because in it the lighting conditions can be controlled without disturbing the user's view. The fiducials used for the system are retro-reflective circular stickers punched out of tape (see Figure 14). There is a strong circular infrared flash attached around the infrared camera. (See figure 15 and 16) This flash illuminates the fiducials without changing the lighting conditions in the visible spectrum. In contrast to the background, the retro-reflective fiducials appear very bright in the image. In addition to this, it allows for very short shutter times ($1/4000s$) which avoids motion blur. The flash is triggered by the shutter of the tracking camera. The trig-

Figure 14: The fiducials are glued on plastic frame as a set of fiducials. The left photo has been taken without a flash and the right one with a flash.

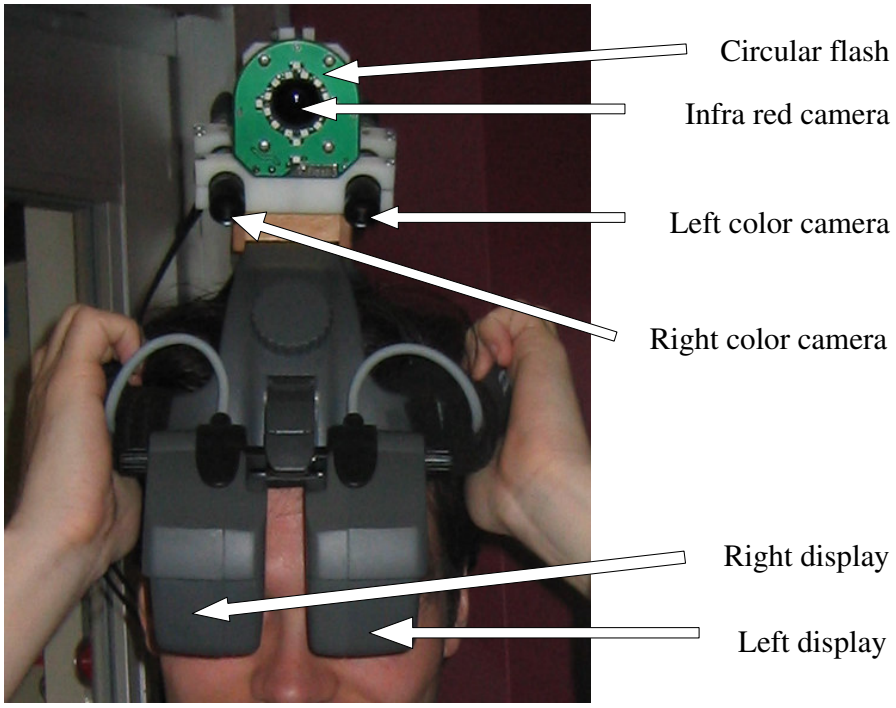

Figure 15: Front view of HMD, flash and cameras

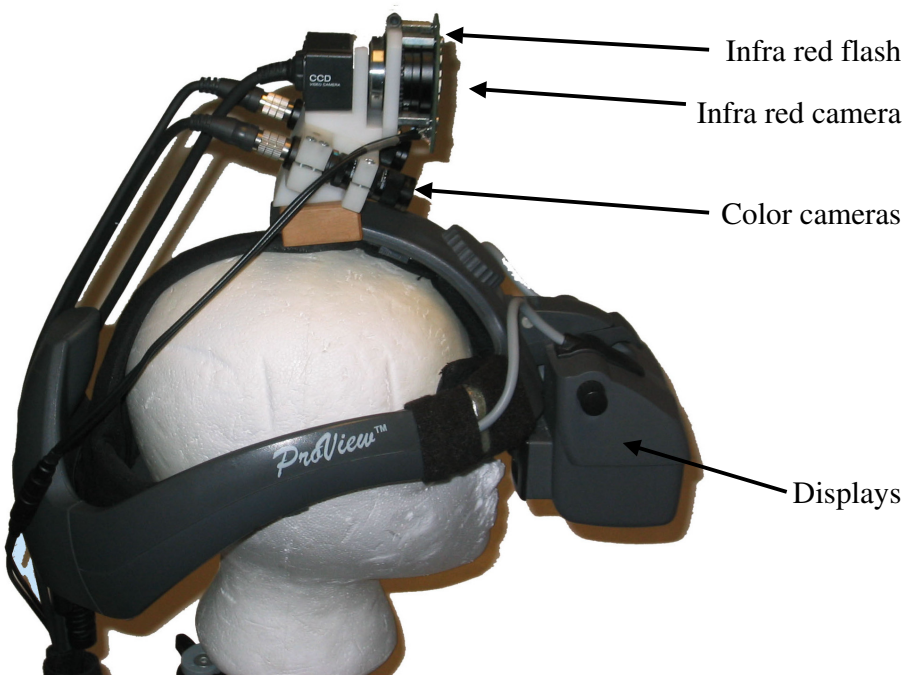

Figure 16: HMD, flash and cameras from the side

gering makes it possible to increase the electric current on the infrared diodes and thus their brightness without causing them to burn out. A description of RAMP is presented in [60], but it is not completely up to date. The system consists now of a single PC but not of three SGI machines.

The cameras of the system have a fixed focus and a fixed zoom. All of the cameras have a focal distance that allows for objects to be sharp in range of the user's arms. The infrared camera has a fish-eye lens that extends its angle of view and therefore its possible range of tracking. The fixed focal length of the tracking camera allows for a very precise estimation of its internal camera parameters at high speed. Precise estimation of the internal camera parameters is time consuming. Therefore it has to be done before the system is in real time mode. This way of handling the internal camera parameters exploits the assumption that all of the parameters including the focal distance (i.e. the zoom) remain constant.

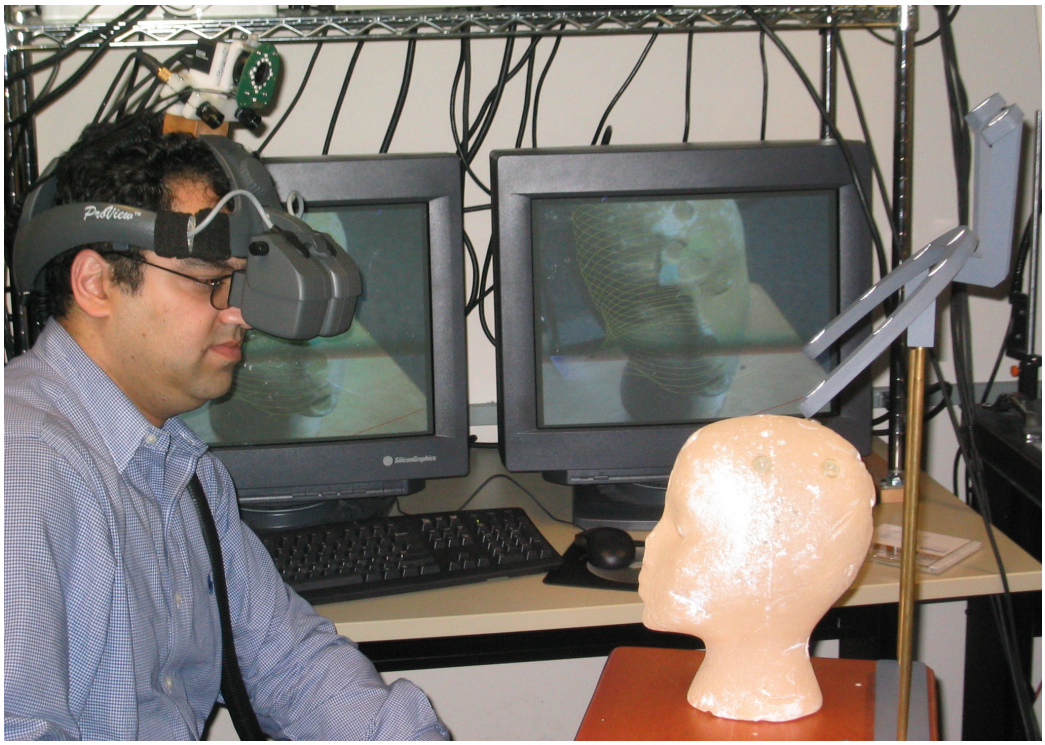

Figure 17: This is RAMP at work. The displays in the background show the images displayed in the HMD. The head phantom is attached to the set of fiducials.

6.3 Reasons for RAMP's hardware composition

For understanding the system and in order to get some insight into problems and trade-offs medical Augmented Reality has to deal with, it is a good idea to find out why the hardware components have been chosen the way they have. In this section we take a look at trade-offs like solving certain problems by software other systems solve by hardware, e.g. identification of fiducials. More about this particular task later on (see section 6.3.3).

6.3.1 Visualization with a video see-through HMD

RAMP's HMD[6] technology has been chosen because of the reasons named in section 5.1. Dependable accuracy can be provided easiest by video see-through HMDs. Neither can eye-to-HMD calibration add an error because this calibration is not necessary, nor a time lag between real and artificial images may disturb the alignment of virtual and real object, since the lag can be set to zero. Furthermore the presentation of real and virtual objects

appears more coherent because both suffer from the limited means of the displays.

6.3.2 Tracking

Tracking hardware has been chosen to meet the requirements (see section 1.2) of high accuracy, dependability, robustness and low costs. RAMP[10] takes advantage of the accuracy of visual tracking. In order to increase accuracy and speed of tracking fiducials are added into the setup. Inside-out tracking has been chosen because it is assumed that a system using this type can be made more accurate than its counterpart. To illustrate the underlying thought: In order to keep tracking errors as small as possible it is important to make the fiducials cover as much space as possible (see section 5.5). In terms of accuracy it does not matter on which side the camera or the set of fiducials are placed. While keeping the space covered by fiducials large, it is a better idea to place the bulky set of fiducials around the patient rather than on the user's head.

Another option for keeping results accurate would be multiple camera views as opposed to RAMP's single camera tracking, and place the cameras in considerable distance to each other. This alternative would be impractical in clinical use, because occlusion of sight of the cameras is likely in an operation room. This alternative would result in less robustness.

Another reason for inside-out tracking is RAMP's ability of tracking tools. As explained in section 3.2.5 the required tracking accuracy for a tool can be maintained more easily with inside-out tracking. An infrared camera is used in order to control light conditions without interfering with the doctor's view. Other high accuracy systems (e.g. BrainLAB's VectorVision [86], NexGen's Optotrak [87], or A.R.T. [85]) track infrared images as well, but they use multiple camera vision while RAMP uses just one camera for tracking. With multiple camera view, it is possible to generate three-dimensional knowledge about fiducials. Therefore, the identification task of similar looking fiducials is just a 3D-3D registration and not, as in RAMP, a 2D-3D registration task. Identification is of course easier with this additional knowledge of depth. A second camera increases costs again, adds weight and needs a certain distance to the other camera. The latter one is the biggest disadvantage. A sufficient distance for RAMP would be about $30 - 40$cm, but it is probably too inconvenient to wear a construction of this size - in addition to the HMD.

6.3.3 Fiducials

The remaining decision is between active and passive markers including their way of being identified (see section 4.8). Active markers have been used successfully for commercial tracking systems like NexGen's Optotrak [87]. Those markers are diodes emitting pulsed light for identification. Thus, identification is not a problem with a small marker size in contrast to passive markers that can only reflect light. Usually visual tracking systems with passive markers include or surround some kind of bit code or symbols for identification [40], [41], [43], [24]. That bit code or the symbols must be extracted robustly which usually leads to bulky fiducials. Bulky fiducials have the disadvantage of being likely to be occluded by themselves, hands, or tools. Furthermore, the bigger fiducials are the fewer fiducials can be placed into an image for redundancy. Thus, fiducials with an uniform shape allow for more robustness and accuracy than those with an identifiable shape.

On the other hand, active fiducials have their disadvantages as well. Since they are active, they need electric currency from batteries or even mains. Batteries add unnecessary weight to tools and they might run flat just when they are needed most. As another drawback it is difficult to disinfect a set of active fiducials without harming them. Furthermore, active fiducial sets are more complicated and costly to produce than their passive counterparts. Last but not least, active fiducials need electric circuits. These might distort CTs[3] or MRIs[9] when inserted into the tube for registration purposes. A recent development in MRI is called cine-mode. In cine-mode just one slice is shown, but it is changing a few times per second. Trying to augment MRI in cine-mode has been one research interest of RAMP which would be impossible with active fiducials.

To summarize, of the four techniques for identification suggested in section 4.8, three are not feasible for RAMP. Adding information like barcodes reduces maximum accuracy, pulsed light needs electric circuits in markers and color coding is impossible in infra red light anyway. The remaining technique that has been chosen exploits geometric constraints of sets of fiducials. This is a particularly difficult task because markers can be occluded, additional markers from other sets can appear, and only 2D data is available since RAMP is a one camera system. The algorithms employed are shown is section 6.4.3 and section 7.7.

Brightness of passive fiducials can be enhanced without drawbacks by an infrared flash. Passive markers appear bright enough in the image when a strong flash is illuminating them. Using a retroreflective material enhances brightness and increases the contrast against the background. A high contrast supports accurate extraction of fiducials. The reflection is, of course, only

strong enough for a short range but the system is meant to work at a short range that can be reached by hands. Consequently, reflection from the flash is sufficient. If the identification problem can be solved while keeping fiducials small; passive fiducials would be the better option.

Circular flat stickers have been chosen as fiducials. They are made of a retro-reflective material to support the flash and distinguish themselves from other materials in the set. There is no difference between any of the fiducials so they can be kept as small as possible. This kind of setting with passive fiducials and infrared flashes is already available in a commercial system named ART [85] and it has been proven to work well. The difference between RAMP's[10] fiducials and ART is the fact that ART prefers spherical fiducials while RAMP prefers flat fiducials. Flat fiducials have a shape more complicated to calculate and they have a limited angle of visibility, but they are cheaper to make. Furthermore, they can be produced more precisely and they can be attached more easily. RAMP can afford to use flat fiducials with a limited angle of reflection because it is a one camera system. Since fiducials have to be visible in both cameras at the same time flat fiducials can limit the range of use a lot. As an example RAMP's fiducials reflect light sufficiently if the system's view has an angle of less than 50° to the normal of the fiducial. If a stereo camera system has a convergence angle of let us say 60°, the fiducial is only visible if the angle between the normal of the fiducial and the bisector of the convergence angle is less than 20° which is already very limiting.

6.3.4 Cameras

All cameras produce interlaced images. This means that for image retrieval odd-numbered lines are scanned first and even-numbered lines are scanned second. Figure 18 illustrates the effect of interlacing for dynamic objects. This technology is widespread among cameras because it reduces flickering in image sequences without increasing the frequency of recording. A higher refresh rate means shorter shutter times which might reduce image quality. Since the user is not supposed to see the image of the infrared camera, flickering would not be a problem. Interlacing must be kept in mind for computer vision algorithms. As described in section 7.4, interlacing raises some problems to be taken care of. That is why for tracking a camera producing non-interlaced images would be more recommendable, but most off-the-shelf cameras only work in interlacing mode. Obviously, a trade-off has been made and a cheaper component chosen.

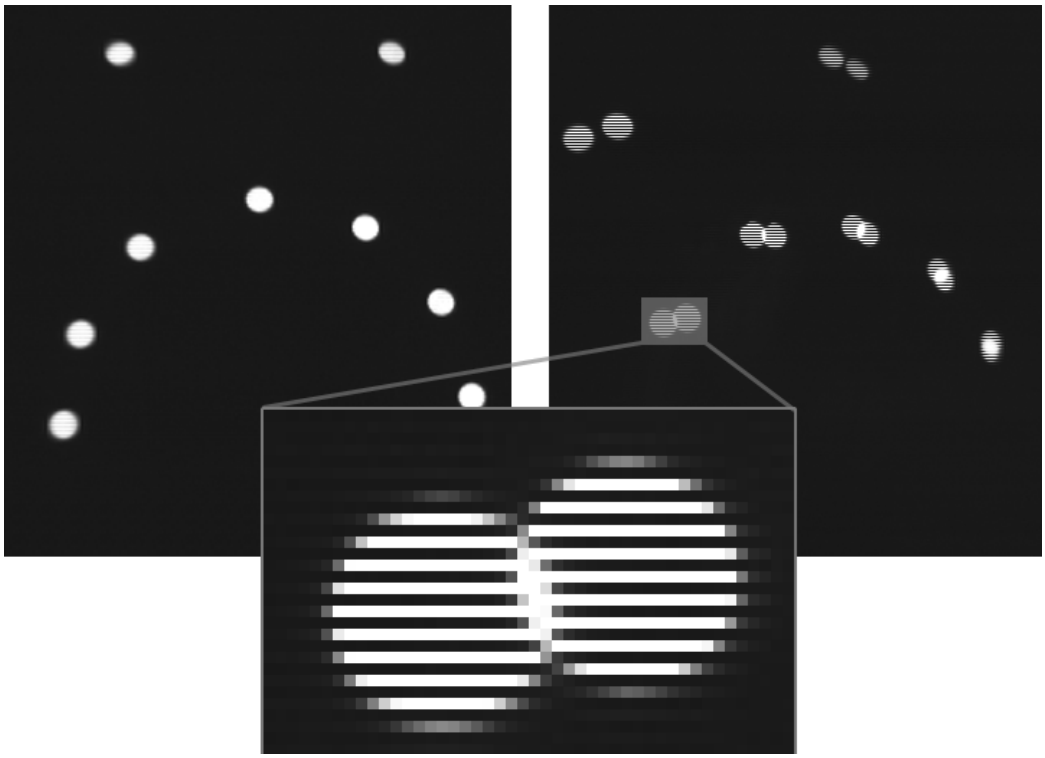

Figure 18: The left hand image shows an object that is not moving. The right image shows the same fiducials moving fast. Interlacing artifacts as seen on the right are not noticed by the eyes if the images are updated frequently enough as in a video sequence

6.3.5 Computational components

Computational load of RAMP[10] is shared on its two pillars of computation: There are two processors and one graphics card. A dual processor PC can distribute the two concurrent software parts of RAMP on two processors to increase the overall speed. The architecture of RAMP as two processes supports parallel computing on two processors. However, it is doubtful if the two processors can increase the performance as can be seen in figure 19, because the computations of server and client are not concurrent but sequential. The second pillar is a graphics card supporting fast 3D computation. Augmented Reality is an application that makes heavy use of 3D computation for its visualization. RAMP's[10] software makes use of OpenGL [42] as a library hardware-supported library.

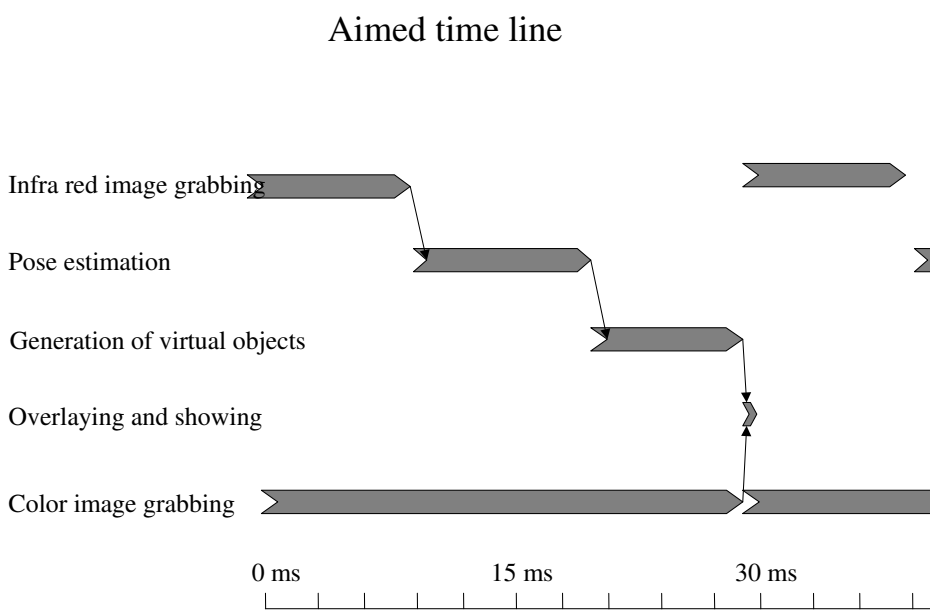

Figure 19: RAMP aims in this time line. The idea is to perform all calculations while the color images are grabbed. This would lead to a refresh rate of about 30 fps (frames per second) and a lag of about 30 milliseconds.

6.4 Description of the existing RAMP software

6.4.1 Two parts of RAMP's software

The architecture of the system is divided into two pieces of software that might even run on two different computers. The part called 'RAMP server' provides the position of the head relative to a marker set that defines the absolute coordinate system. Additionally, it provides the position of tools with fiducials attached. The server does not do anything but taking the grabbed image from the infrared camera, extracting its relative position to the sets of fiducials[4] and sending this data to the RAMP client. RAMP's server does the tracking part.

The other piece, namely 'RAMP client', takes the 3D position information from the server, and renders the objects into one view for each eye. Each of these rendered images is combined with the real view from the two color cameras. Rendered virtual objects are simply overlaid onto their real counterparts. This might result into misleading visualization if virtual objects are

behind real objects. RAMP does not gather three-dimensional information about the real environment. There are algorithms to gain depth information about the environment by stereo vision [10], [28],[9], [29] but it takes too much time to compute and the results are not very accurate either. As a compromise, RAMP's users have to do with misleading depth cues from wrong occlusion if a real object is in front of a virtual one but enjoy a faster system. Time is a very critical resource in this real-time system. In order to maintain the impression of natural objects behaving virtually an update rate of about 30 frames per second must be provided (see section 5.8). Therefore, we have just 30 milliseconds for grabbing, retrieving the positions, visualizing the objects, and combining them for showing on the display. That is also the update rate of the framegrabber[5] of the cameras. This means that every 30 milliseconds a new image is available for augmentation. If the time consumption of tracking and visualization is higher than that, frames must eventually be dropped, latency time increases and the update rate decreases. This makes the images appear less natural and even worse, it might lead to headache or other symptoms of cyber sickness as described in chapter 2.3.

A fast and effective method for avoiding problems with the depth cue of occlusion is overlaying transparent objects. Transparency helps the brain overcome occlusion as a strong depth cue.

As this thesis is about tracking for Augmented Reality, I would like to focus on the tracking part of the software.

6.4.2 RAMP and software engineering

In the very beginning it has not been clear what problems would occur and what specific tasks RAMP[10] could actually support. After all, RAMP is a research project and its outcome was not totally clear. A major part of this research project is in fact about finding a concrete application helpful for actual use. In this case, the application depended on what RAMP's team would be able to build. Software engineering [18] projects start usually with requirements elicitation, but requirements are difficult to define if the application is not clear. Software engineering aims at high quality software as highest priority while computer research projects aim at rapid development of many kinds of possibilities. These possibilities may only occur during implementation, so a tight design might not support the research. Even worse, software engineering is a way of developing software that helps not to change the application or problem domain because of new developments. But changing the application to find a good one is in this special case desirable. Hence, RAMP has not been developed with software engineering tools. As a side effect, a lot of changes and extensions of RAMP's code made it difficult

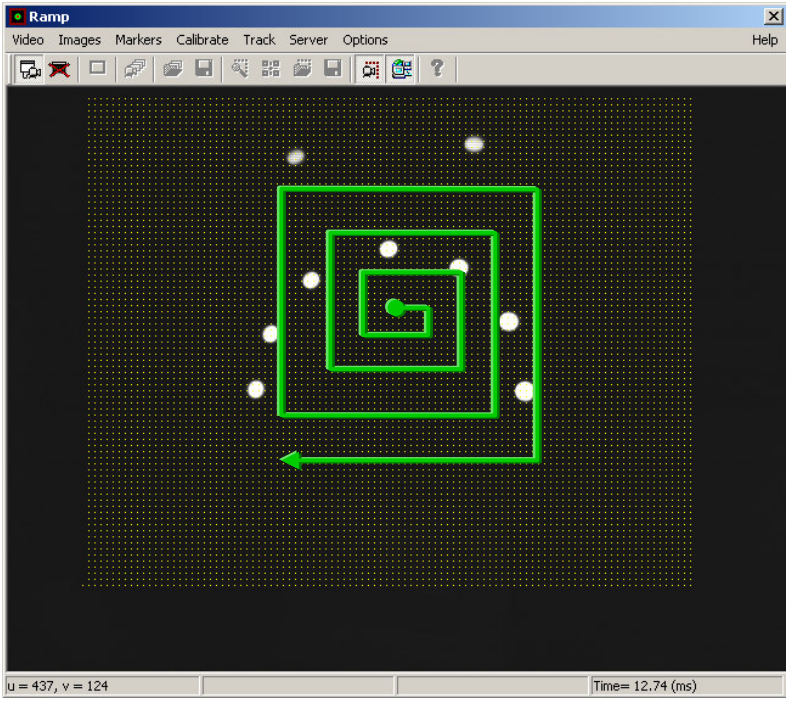

Figure 20: Fiducials are being searched for starting from the center of a pixel grid, moving onwards circularly

to maintain. Therefore a new object design of the tracking classes has been implemented (see 7.1) which makes it easy to recombine different parts of tracking.

6.4.3 Workflow of RAMP's tracking

First of all, fiducials have to be found in the image. This step used to be done by checking for each n-th dot on a grid whether it exceeds a certain threshold. This used to be each 16th dot. Figure 20 illustrates the search. Improvements of this can be found in the section 'New Features' 7.2. Using an easy region growing algorithm, the rough size of each spotted fiducial has been estimated. Features of these fiducials must be extracted from the image at sub-pixel accuracy. RAMP[10] makes use of moment extraction of the fiducial's area (see section 5.3). This is done in the estimated rectangular area around each fiducial. Despite interlacing[7] both half-images are treated as one image for maintaining vertical accuracy. Changes of this treatment are described in section 7.4 and 7.3. The measured fiducials must be identified for calculating the perspective n-point problem. This task could be performed

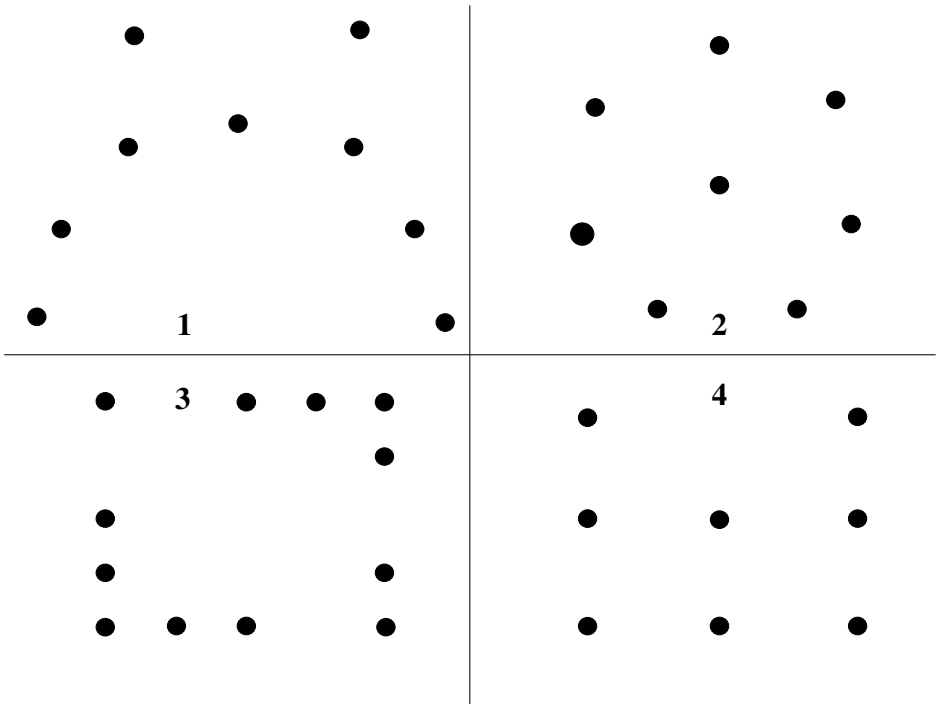

Figure 21: Four sets of fiducials. The sets in the top row have a distribution in space while the ones in the bottom row are placed in a plane. The sets on the left are examples for sets for world coordinates and the ones on the right are attached to tools.

for two sets of fiducials. One set is considerably bigger and provides world coordinates while the other makes it possible to track a tool. Changes to track more than one tool are described in section 7.8. Unfortunately, for each new shape of a set of fiducials a new heuristic had to be invented, or the other way around: For each new task with new constraints a set of fiducials had to be conceived fitting the algorithm that identifies fiducials. Figure 21 shows four examples of sets of fiducials for RAMP. Each of these sets needs a certain heuristic to identify its fiducials.

1. The first set defines the world coordinates. It is assumed that the three markers on the top are visible and furthermore on top of the image. With these three markers a rough transformation is calculated in order to identify also the other fiducials as well. The advantage of this heuristic is that the markers may have a spatial distribution and all of the markers except for the top three may not be visible without interfering the systems's operability. As a disadvantage, the initial

assumption can be too restrictive.

2. This set is for a tool. There is one fiducial that is bigger than the others. It is assumed that all of the fiducials are visible. The middle one remains after projection in the middle. The other the fiducials are enumerated clockwise starting at the bigger fiducial. As drawbacks, there are the rigid shape of the set which must be circular and the necessary visibility of all fiducials.

3. This set of fiducials is in a plane. Thus, for the fiducials on each side of the box the cross ratio (see 4.6.2) can be calculated. Only two of the four lines must be visible for tracking. As a major disadvantage objects should be augmented only into the plane of the set for numerical reasons (see section 5.5).

4. Last but not least, there is the planar set for a tool. This set has the disadvantage that no fiducial is allowed to be occluded and furthermore only augmented objects should be in the plane of the set, as for the set before. This set has been used for augmenting ultrasonography images at the probe where it has been taken.

A solution for this problem for all kinds of sets of fiducials with arbitrary invisible fiducials, an arbitrary spatial distribution, is described in section 7.7. After identification, a rough estimate of the perspective n-point problem is calculated. This is done employing an analytic approach in order to gain a starting value for the iterated optimization, which is made us of to compute the exact transformation. The transformation is sent to the RAMP client if the error is not above a certain threshold. The error is calculated by projecting the model into the image plane in accordance with the transformation Then, the distance of each fiducial to its estimated location is determined. Hence, the error of the tracker is given as an average deviation in pixels. The error in terms of rotation and translation remains unknown.

7 New Features

As the practical work done for this Diplomarbeit is about extending an existing system I would like to point out which parts are my responsibility. Hence, they are separate from section 6. All of the software was developed in C++ [32] using [33], [34] and [89] to keep the code as maintainable and bug-free as possible.

All of the following additions to the system fulfill the real time constraint of the system. This has not only been tested successfully under laboratory conditions but also in a clinical test. These algorithms are fast enough for the system to meet the desired time line as shown in figure 19. Since this is true for all of the presented additions, it will not be presented in the results in each of the subsections. The exact speed of each algorithm depends on the hardware in use. Therefore the complexity might be more interesting, but I would like to point out that the presented algorithms can be used with today's off-the-shelf hardware for real time visual tracking at $(30Hz)$.

7.1 Redesign of tracking classes

Several tracking classes have been implemented with different algorithms for fiducial extraction, marker set identification, and pose estimation. Since these classes did not share a common platform of interfaces, a small framework for tracking has been implemented in order to change or recombine algorithms easily. There are four major tasks associated with tracking:

1. **Finding markers in the image**. This is about low-level computer vision. Finding markers means marker segmentation and feature extraction.

2. **Distinguishing markers by their objects**. This is just in case there are markers that belong to different objects. Distinguishing markers by their objects can reduce complexity for algorithms solving the tasks of identification. Distinguishing marker can already imply the identification of each marker. It has to be done for all of the extracted fiducials as a whole.

3. **Identifying markers on each object**. Identifying markers is about sorting markers that have been found in the image according to their model. This order is in general not the same as the order that low-level algorithms provide. This task has to be done for each model.

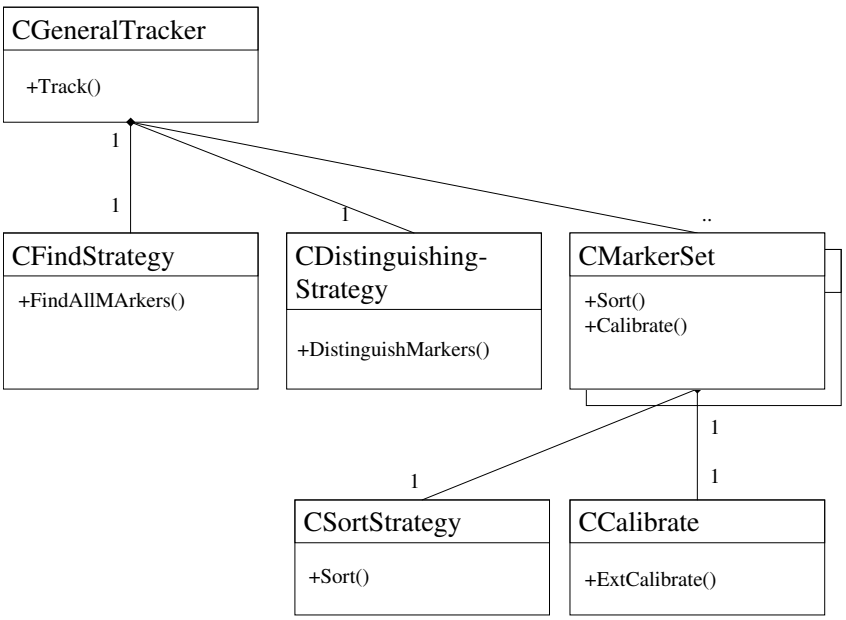

Figure 22: Class diagram of the parent class for tracking shows just aggregation and public functions.

4. **Pose estimation**. Last but not least, exterior camera calibration is done by solving the perspective n-point problem for the extracted and sorted markers. This is computed for each set of fiducials.

These four independent tasks can be separated into different classes. For an easy combination of these tasks a strategy pattern as described in [18] has been chosen. The abstract class for general tracking possesses strategies for the first two tasks (for finding `CFindStrategy` and distinguishing `CDistinuishStrategy`). Both of the other tasks are dependent on the object to track to increase versatility. Hence, the class for general tracking possesses an object for each marker set to track (RAMP class `CMarkerSet`) that possesses one strategy for identifying markers (`CSortStrategy`) and one for external camera calibration `CCalibrate`. Inheritance diagrams in figures 23 and 22 illustrate the relationships between classes. Every tracking class inherits a function from `CGenaralTracker` that is similar to the one in pseudo code given below.

```
Track(image){
  rawMarkers = FindMarkers(image);
  markerSets.markersIn2D = DistinguishMarkers(rawMarkers);
```

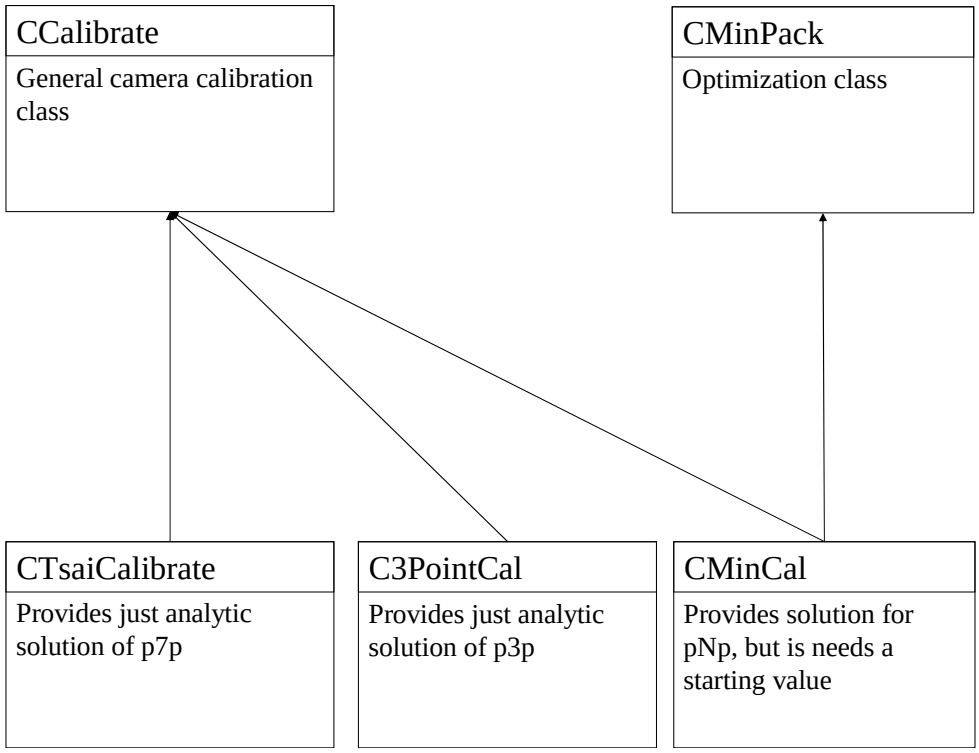

Figure 23: Class diagram of the parent class for calibration. Shows only RAMP's basic classes for calibration. RAMP uses combinations generated by aggregation of the basic classes, e.g. `CTsaiCalibrate` and `CMinCal` after another produces the calibration suggested by Tsai et al. [7]

```
  for n= 1 to markerSets.length
    markerSets[n].SortMarkers();
    if markerSets[n].wellSorted {
      results[n] = markerSets[n].Calibrate();
    end; //if
  end; // for
  return results;
}
```

Each function makes use of the strategy object that has been instantiated. If we want to change the behavior of a tracking class we just need to change the strategy objects but nothing else. Note that this concept is generally independent of the fact whether markers are artificial (fiducials) or natural. That is why the term 'marker' has been used. The C++ function that is

used in RAMP[10] is actually quite similar to the one in pseudo code, but it involves some `if`-statements to catch exceptional behavior, so the original code would confuse rather than clarify at this place.

7.2 Efficient marker detection

7.2.1 Description

Moments of areas (as described in section 5.3) have been chosen as a sub-pixel accurate feature for RAMP's[10] visual tracker. First, these areas have to be segmented into one segment for each fiducial. This used to be done by a circular search from a seed point (see figure 20). Whenever a pixel has been spotted that belongs to a fiducial which is determined by a threshold, a region growing algorithm is used to determine a rectangular area covering all of the fiducial. Unfortunately this way of segmenting has several drawbacks. Firstly, by searching circularly the algorithm moves through the image not only from left to right and back but also up and down making cache misses very likely. This considerably slows down memory access. Secondly, if each pixel was probed, overhead would increase considerably for finding out whether a new detected pixel is in a region that has already been detected. Therefore, not each pixel is probed individually. Instead, each n-th pixel on the circular route is examined, causing small fiducials and non-solid fiducials to fall through the grid. Non-solid fiducials are planned to be used for distinguishing between fiducials of different objects. Hence, something had to be changed.

7.2.2 Approach: Modified Connected Component Analysis

As a solution, a modified Connected Component Analysis (as suggested in [22]) has been used for segmentation already providing rough values of the moments of the area. The algorithm uses two arrays `currentLine` and `previousLine` with length of the width of the image. The algorithm iterates line by line through each pixel in the image. If a pixel has a gray scale value beyond a certain threshold it determined to belong to a marker. Figure 24 illustrates the five cases that can occur if a pixel is found that belongs to a marker. The algorithm is given is pseudo code and is explained afterwards.

```
findMarkers(image, highThreshold, lowThreshold){
  currentLine[image.width+1];
  previousLine[image.width+1];
  dynamic_array markers;
```

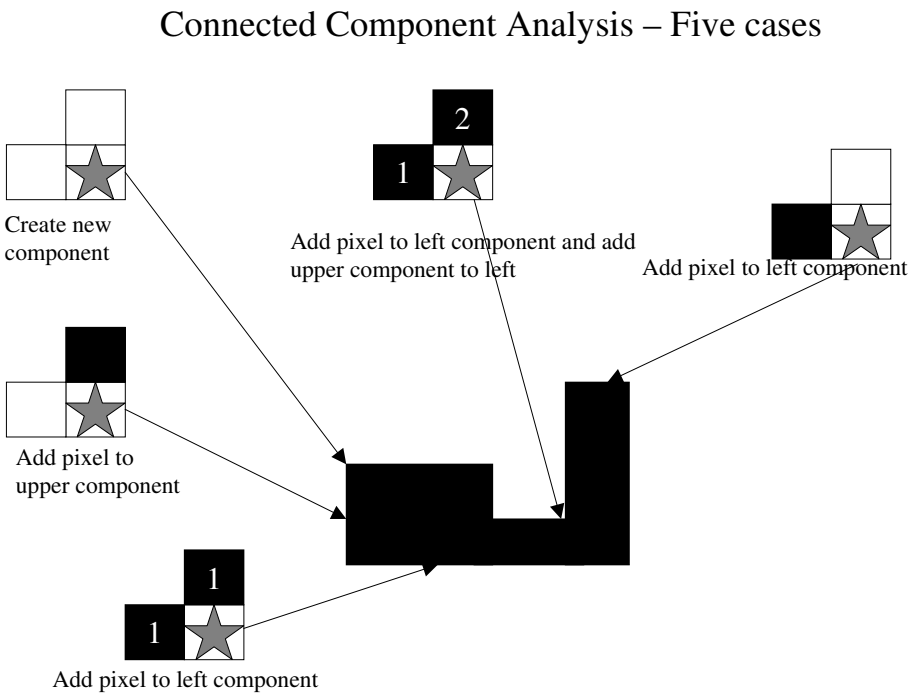

Figure 24: Five cases that may occur during iteration through the image if a pixel that belongs to marker is examined (star)

```
threshold = highThreshold;

for i = 0 to image.height
  for j = 0 to image.width
    if image[i,j]>threshold
     // hysteresis thresholding
     threshold = lowThreshold;
      value = image[i,j];
      markerOfLeftPixel  = markers[ currentLine[j-1]  ];
      markerOfUpperPixel = markers[ previousLine[j-1] ];
      if thisLine[j-1] != -1
        markerOfLeftPixel.addPixel(i,j,value);
        // assign number of left pixel to
        thisLine[j]= currentLine[j-1];
        if previousLine[j]!= thisLine[j-1]
```

```
            // merger of two markers
            mergeMarkers (markerOfLeftPixel , markerOfUpperPixel);
          end;
        else   // if left pixel does not belong to a marker
          if previousLine[j] != -1
            addPixel(markerOfUpperPixel, i,j,value);
          else // if both, left and upper pixel do not
               // belong to a marker
            markers.createAndAddNewMarker();
            // assigns current pixel to the last created marker
            currentLine[j]=markers.length;
          end;
      else // if current pixel does not belong to a marker
       thisLine[j]=-1;
       threshold = highThreshold;
      end;
    end;
    swap(thisLine, previousLine);
  end;
  return markers;
}

addPixel(marker, i, j, value){
  marker.area = marker.area + value;
  marker.sumX = marker.sumX + value * j;
  marker.sumY = marker.sumY + value * i;
  marker.minX = min(marker.minX, j);
  marker.minY = min(marker.minY, i);
}

mergeMarker(marker1, marker2){
  marker1.area = marker1.area + marker2.area;
  marker1.sumX = marker1.sumX + marker2.sumX
  marker1.sumY = marker1.sumY + marker2.sumY
  marker1.minX = min(marker1.minX, marker1.minX);
  marker1.minY = min(marker1.minY, marker1.minY);
}
```

Explanation of the algorithm: If a pixel on the left of the current pixel already belongs to a marker, that pixel is connected to it in a four-neighborhood.

Thus, it is added to that marker and the current pixel is marked in the array of the current line to be part of that marker. If the marker above also belongs to the same marker no further action has to be undertaken, however, if it does not belong to the same one, this means that two different components have been found that are connected and thus they are one component. These parts of one marker have to be merged.

If the left hand pixel does not belong to a marker, the upper pixel might belong to the marker and in this case it would be added to that one. If not, a new marker has been found, an object for it is created and added to the list of markers. The current pixel is marked in the array as belonging to the last marker created.

To make the detection of markers more robust, a hysteresis thresholding has been added to the algorithm. This means that if the last pixel belongs to a marker a lower threshold is used. If the last pixel does not belong to a marker, the higher threshold is applied. This reduces the chance of a part of a marker being separated just by the influence of noise. If this behavior is not desired, of course, setting higher and lower thresholds to the same value stops hysteresis thresholding.

7.2.3 Results

The complexity of this algorithm is benevolent. For an image sized $n \cdot m$ its computational complexity is $O(nm)$ because the highest complexity in each branch is constant. Adding a pixel to a marker and merging two markers only results in a constant number of additions. Creating a new marker is deemed to work in constant time, too, even though it is stored in a dynamic array. The maximum number of markers is known in advance, so the algorithm can allocate enough memory for the dynamic array in advance. Therefore the computational complexity is independent of the number of markers which stands in contrast to the old algorithm. Linear iteration from left to right line by line through the image and therefore through the memory supports strategies of the CPU cache and paging strategies of the operating system.

As opposed to the original Connected Component Analysis, the image (or memory of the same size) need not be overwritten. The use of memory is limited to only two static arrays with the length of the image width plus one. This extra piece of memory is needed to stay within the boundaries of the arrays in the first column without an extra `if`-statement for checking boundaries.

7.2.4 Discussion

This way of finding markers provides a very fast analysis of the image without neglecting any of the pixels. It robustly and precisely locates a rectangular boundary of the markers. The zeroth and first moments of the area and its centroid are measured, too, at very low computational costs. The analysis provides results for solid fiducials that as good as for ring-shaped ones.

Unfortunately, the measurements of moments are not as accurate as possible because pixels with a gray scale value under the threshold are not taken into account. Grayscale moments are considerably more accurate. To exploit gray scale values other algorithms have to follow.

7.3 Bias estimation and masking

7.3.1 Description

As explained in 5.3 an exact estimation of the brightness of the background is necessary for an accurate gray scale moment extraction. If the bias is estimated in wrong way the whole measurement is wrong. It is a good assumption that the bias changes through the image since the background is dark but not necessarily black. Different objects can still be seen causing the background not to be constant. Additionally the camera produces darker and lighter stripes in the image for no obvious reason. For both, see figure 25. Therefore the bias must be estimated locally for each marker. A common idea is to employ the border of the extracted area for the estimating the bias. Therefore bias estimation and region extraction go together. The old algorithm for bias estimation only examined pixels on the rectangular boundary box around the segmented markers. The problem with this approach is displayed in figure 26 on the left. The corners of a rectangular boundary box might overlap with another marker which would raise the value of the bias resulting in a wrong measurement.

7.3.2 Approach: Estimation from pixel on a round boundary around fiducials

The solution to this problem is a round boundary box as can be easily seen in figure 26 on the right. There is the problem that the fiducials are circular but their perspective projection is an ellipse. Morphologic dilatation with a diamond-shaped 9×9-mask has been used to maintain a distance of one pixel from the border of the thresholded marker. For more than one pixel the procedure is repeated instead of enlarging the mask. This has a lesser cost of computation. The acquired effect is the same as for a big diamond-shaped

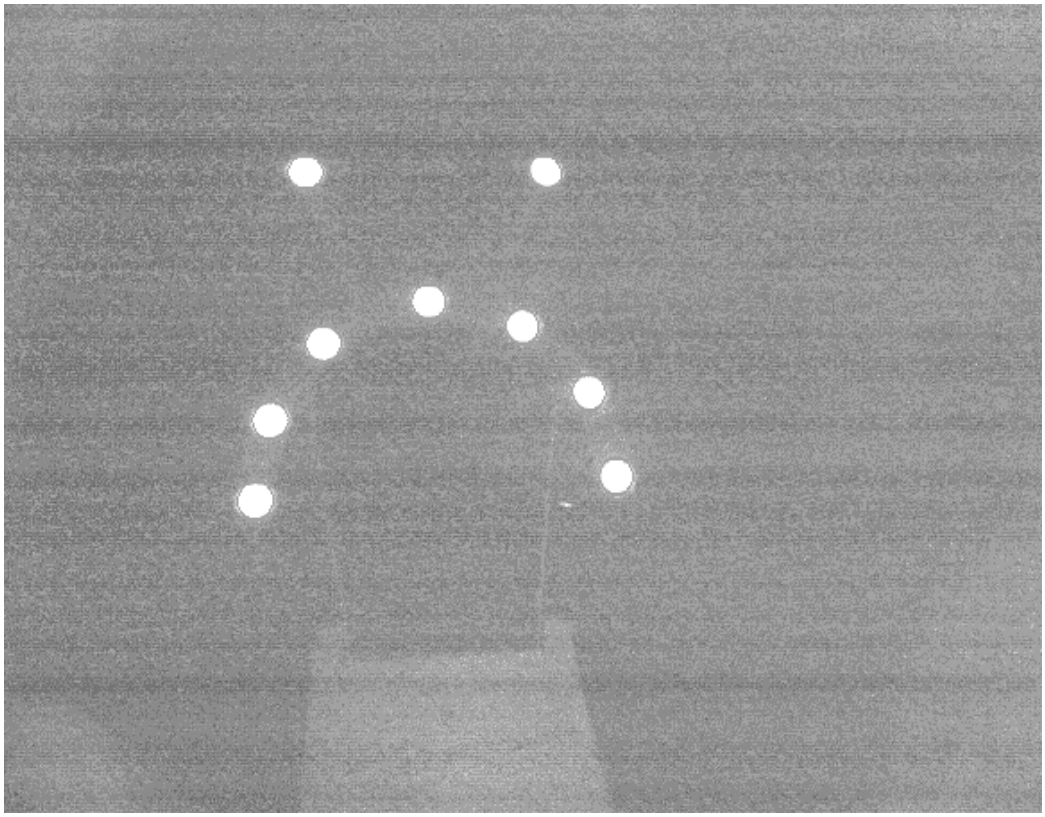

Figure 25: For this image, the dynamics have been changed to show that the background of infrared camera images are neither black nor globally in the same gray value. Note also the horizontal stripes which are camera artifacts

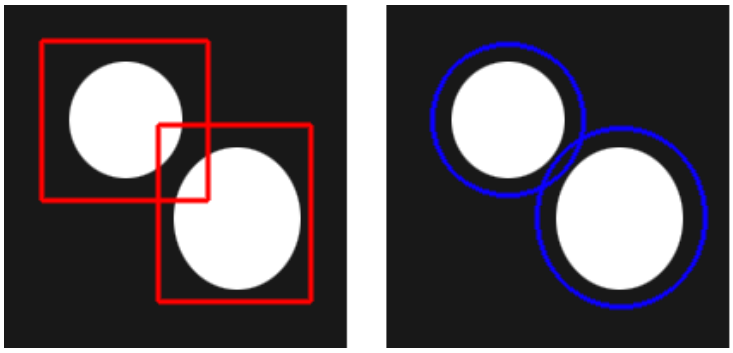

Figure 26: The left illustrates the old way of bias extraction. If markers are close together the bias will be estimated wrongly because the corners of the rectangle overlap with the other marker. On the right there is the solution to it. The boundary box is round as is the marker.

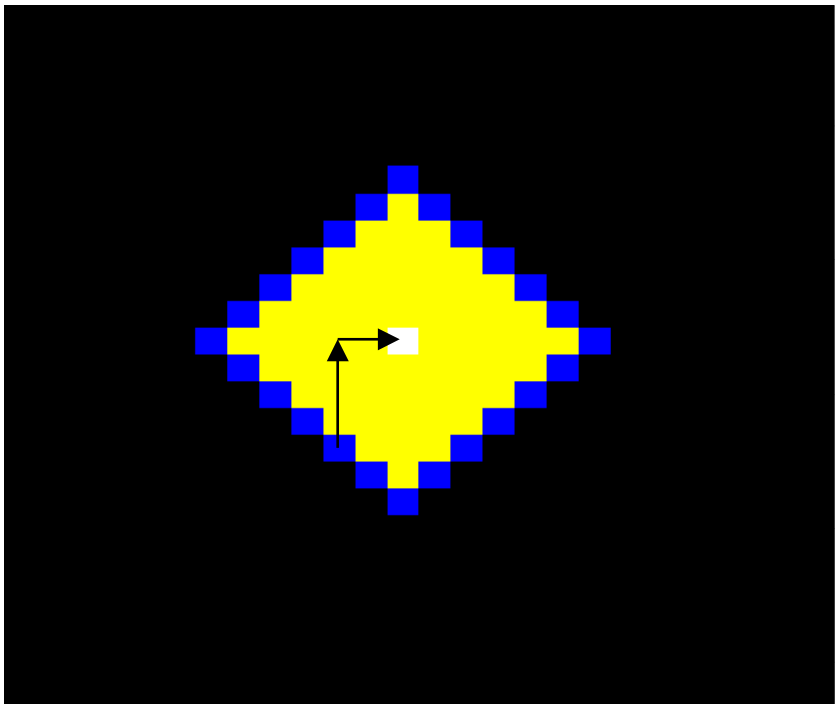

Figure 27: To show the impact of six repeated dilatations only one point in the middle has been dilated several times. In summation norm, blue pixels have exactly six pixels as a minimum distance to a white pixel. Yellow ones have less than six pixels distance to a white pixel.

dilatation. In other words repeated dilatation can conclude what distance d in pixels to its nearest thresholded pixel in the summation norm. The summation norm answers the question how many fields the castle in a chess game needs to cross before it reaches its target.

7.3.3 Results

The algorithm has a time complexity of $O(n \cdot d)$, where n denotes the number of fiducials and d denotes the distance of a boundary point to the extended boundary. The average size of each fiducial $i \cdot j$ is expected to be constant. The results of these round boundaries were not the expected ones, but for future work this subsection is nevertheless interesting. I simply ran out of time. The reason why this way of finding regions does not work optimally is shown in figure 28. The bias is calculated based on the rounded boundaries, so it is probably quite correct. Unfortunately, if the markers are very close together, the centroid of the region selected with this algorithm is not the

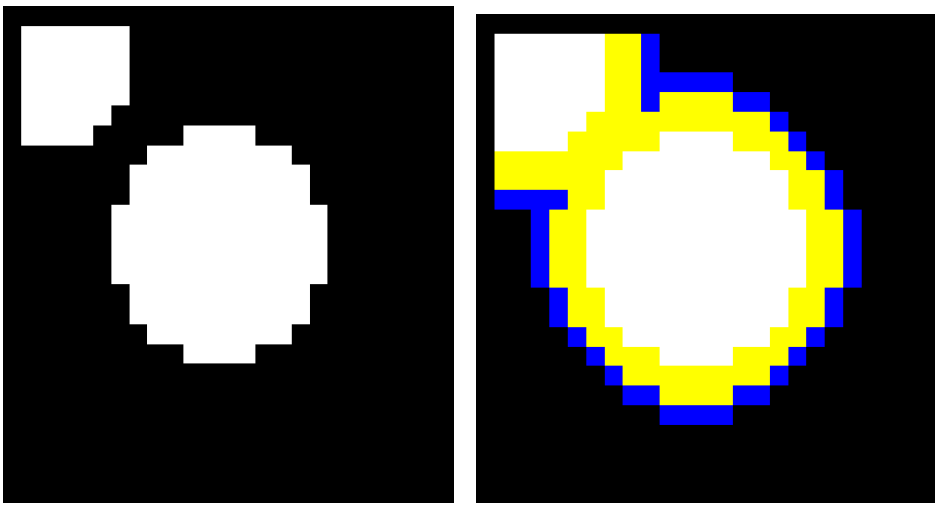

Figure 28: The left hand image shows an extracted rectangle for which the region and the bias is going to be determined. The right hand one is the same image as on the left, but the white and yellow pixels indicate the region that is taken for calculating the moments. Blue pixels indicate the border the bias is taken from.

middle of the ellipse. Hence, the extracted centroid provides an incorrect center of the ellipse in this special case. This is exactly what we wanted to avoid. Eventually the algorithm does not do anything bad, but it does not help as much as desired. There are still problems if fiducials are getting too close.

7.3.4 Discussion

Even though the results look like a dead end of the idea, we can get around the problem. If the algorithm is combined with a CCA[2] to determine the pixels belonging to the detected marker before dilatation and dilate these pixels instead of all pixels beyond a threshold, it will probably provide the desired result. An additional CCA per marker would add a complexity of only $i \cdot j$ per marker, so the complexity would not change.

As another suggestion, the semi-axes could be estimated by a rough esti-

mation of the second degree moments, the same way we obtained the centroid. We do not get these in the first image processing step in which only the zeroth and first degree moments are calculated. With an estimate of the ellipse we can precisely determine an ellipsoid region with the marker in the middle. A CCA as described above will be necessary for this way of producing a round boundary as well. The complexity of this algorithm would be independent of the distance to the marker. Also, distances could be set to fractions of pixels.

7.4 Merging information of interlaced half-images

7.4.1 Description

The infrared camera produces interlaced[7] images. When the head is moved quickly one half-image is much different from the subsequent one. As a first problem, both half-images have to be processed separately. Figure 29 shows that fiducials might appear to be two separate fiducials. This occurs only when the head moves very quickly. Not as obvious is the problem that processing the image not taking into account the time shift between even and odd lines might result in points not having any resemblance to the real fiducials at any point of time. Figure 31 shows the problem. This phenomenon already occurs with differences as small as a few pixels between two half-images. This leads to grave errors. For this reason the old system stopped augmentation if the head moved too fast.

7.4.2 Approach: Merging image information only if head movements are slow

As a new way of treating the interlaced images each interlaced image is processed separately. By handling half-images the resolution in vertical direction of each image is only half as good as before. When using only one half-image, jitter increases considerably because of the lost accuracy in measurement. One solution could be full tracking in each of the half-images and bringing the results together. First, the time for tracking would almost double because most of the time is spent on marker identification and exterior camera calibration, so it is not be a good idea. Second, the outcome is not clear. Calculating two transformation matrices together may provide a different result than measuring more accurately in the first place. Consequently, we opted for both old and new solution combined. If RAMP[10] finds dots in both half-images close enough to each other (threshold is currently one pixel) they are merged as if identified as one. This happens only if a match can

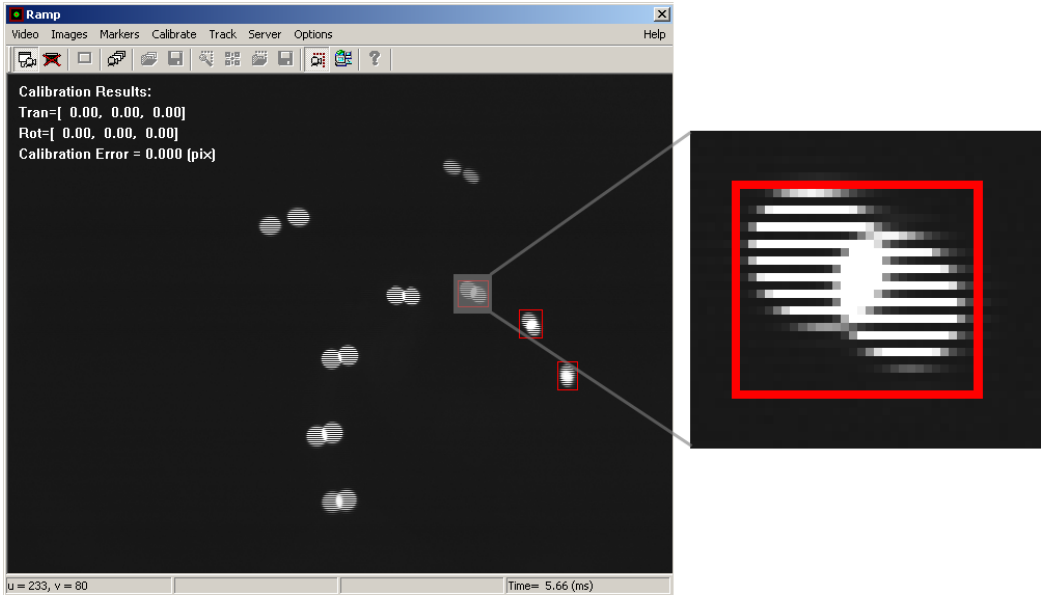

Figure 29: This image shows the old way of merging information of interlaced half-images: They are regarded as one non-interlaced image

be found for all of the fiducials. The last rule has been conceived of because otherwise one part could be merged while the other is not which would cause jitter and provide inaccurate results. If the head is moved quickly now, only one half-image is taken. The lack of accuracy and the increase of jitter is not a problem because the presented images are moving quickly as well. If presented images are not moving much, the eye of the user can determine jitter much more easily than in images moving quickly.

7.4.3 Results

The combination of merging when movements are slow and not merging when movements are quick works very smooth. Now it is not possible anymore to make RAMP[10] lose track simply because of quick movements. Using just one half-image when moving quickly an undesired side effect. The eye merges both presented half-images into a 'middle version'. We can only obtain positions from in between two of those 'middle versions'. One half-image is from a point of time that is too early and the other is from a point of time that is later than our brain deems the video image to be. Thus, we have to make the decision whether to let the superimposed image be slightly behind or a little ahead of the video images.

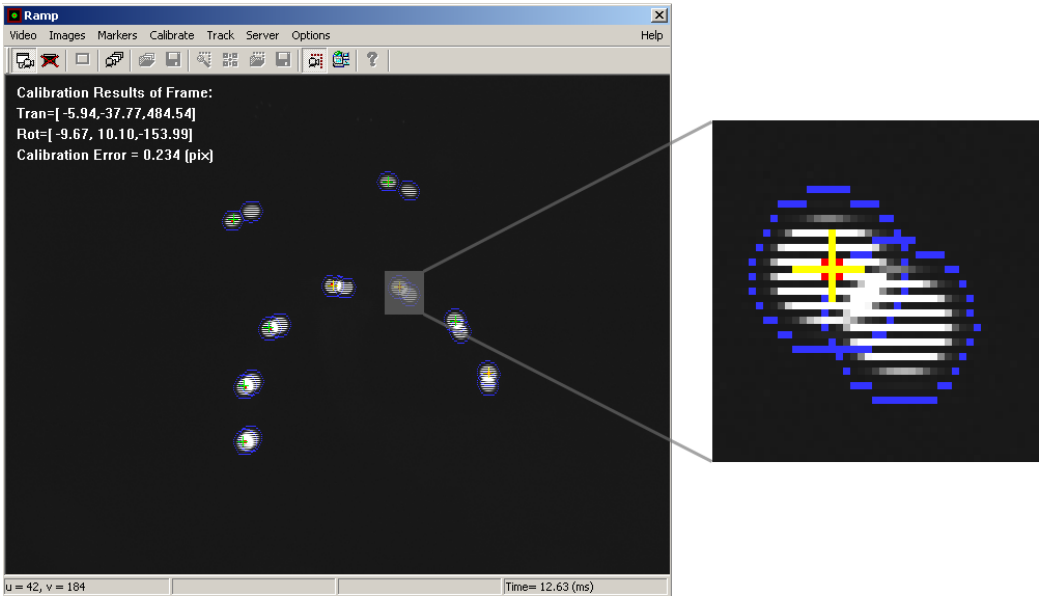

Figure 30: This is the new way of treating interlaced images. Boundary boxes are drawn and evaluated separately in each half-image. The red dot indicates the measured centroid of that fiducial in one half-image. The yellow cross indicates the expected center of this fiducial as estimated by the identification algorithm

7.4.4 Discussion

Combining half-images only when movements of the head is slow has increased the robustness a lot without increasing jitter. Even though doctors might not move hastily during medical procedures, this increases acceptance of the system because it gives the impression of a system that works all the time even under extreme circumstances.

In the future it might be of interest to produce interlaced artificial images to be superimposed, thus creating the impression of smoother, more natural virtual looking object.

7.5 Detecting partly occluded fiducials

7.5.1 Description

RAMP's[10] fiducials cover numerous pixels in the image recorded by the tracking camera. Therefore any partial occlusion will change the center of the fiducial in the image. This corrupts the measurement (see figure 32, 33) and hence the pose estimation. Unlike other fiducial-based tracking systems there

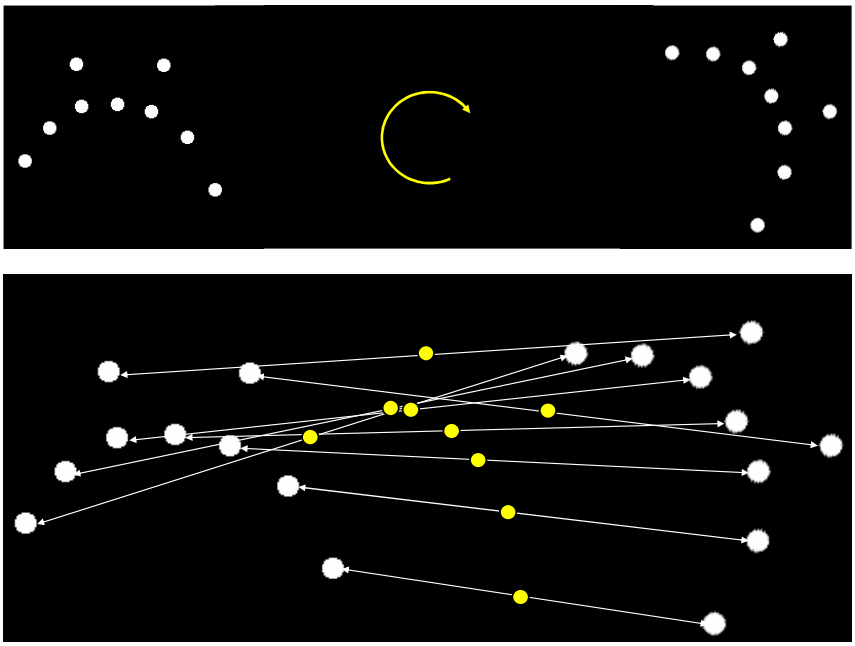

Figure 31: This figure illustrates that interpolating between two images by choosing the center of corresponding dots does not necessarily result in a projected version of the model, but in a distorted one. This distorted set of fiducials (yellow) has no resemblance to the real set.

is no bit code or symbol attached to the fiducials to could tell algorithms that the identification failed. It is a challenging task to detect partial occlusion of fiducials, hence different approaches have been tried. First observation: A partially occluded fiducial in the image produces an outlier for the optimization of the pose estimation. Unfortunately, the optimization does not necessarily identify the wrong one. Taking into account only unoccluded fiducials produces an optimization without outlier. Second observation: Any partly occluded fiducial occupies a smaller area in the image than an unoccluded marker. Of course, occlusion does not increase the size of the fiducial. The majority of fiducials are assumed to be unoccluded. This leads to the idea of computing a rough pose estimation, and the size of a fiducial we expect to be in the picture. If a fiducial appears smaller than expected, it can be sorted out. The algorithm did work out to a certain extend, but unfortunately, it could only robustly detect a partially occluded fiducial if most parts of it were occluded. Another drawback was the fact that this algorithm could detect only one partly occluded fiducial with a certain robustness. The reason is

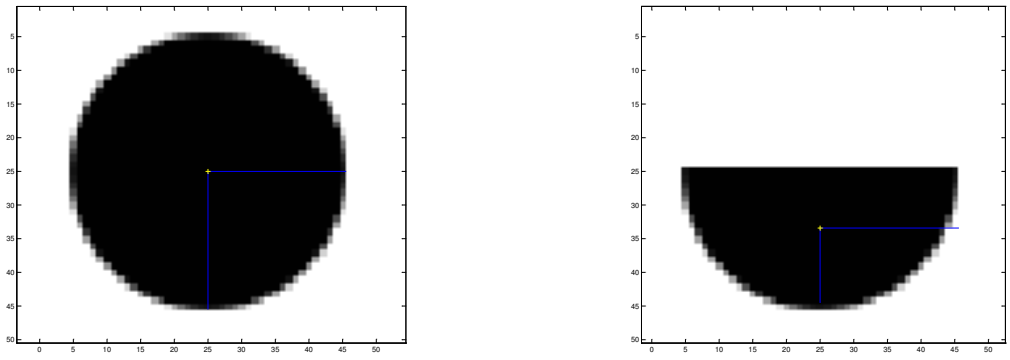

Figure 32: Circle and its center of mass and its semi-axes

Figure 33: Half-occluded circle, its center of mass, and its estimated semi-axes

as follows. The size of the fiducials in the image might appear to be altered due to the effect of overshining (see figure 34). This effect is caused by too much light hitting a camera's photo sensor - i.e. overexposure. Factors that can contribute to or cause this effect are the distance between fiducial and camera, the brightness of the flashlight, the sensitivity of the photo sensor, the material the fiducial is made of, and the angle at which light is reflected from the marker, to name the most important ones. In order to deal with this effect, we compared the size of each fiducial in relation to the others. This was done assuming that the effect of overshining was similar for all individual fiducials. This assumption introduced more sources of significant errors than expected. By placing the set in a certain position and at a certain angle, the error can be made as grave as one likes - or rather does not. This may happen if an extremely overshining fiducial in the foreground or an especially dark one in the background produces false positives. To make things worse, if two or more fiducials were partly covered the average size would decrease and bring these fiducials back into the relative bounds of unoccluded ones.

7.5.2 Approach: Comparison of anisometry of measured and expected fiducials

Learning from this first approach, we had to find a characteristic that is not affected by overshining. Third observation: In most cases, partly occluded fiducials have a different shape compared to their original. It is very unnatural to accidentally occlude only parts of the marker and arrive at the same

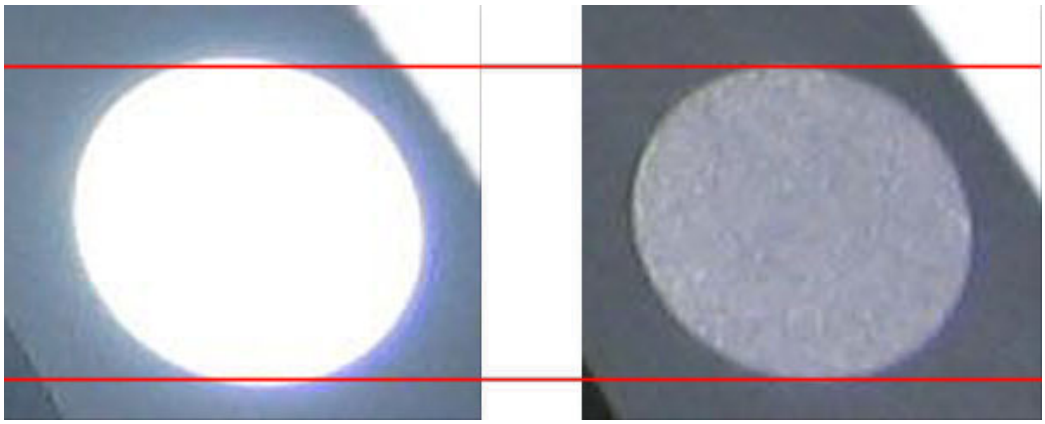

Figure 34: Images showing the same retro-reflective fiducial photographed with and without a flash. The red line is aligned with the top and the bottom of the correctly illuminated fiducial on the right. Bright regions in an image appear larger than their counterparts due to overexposure. This effect is called overshining.

shape of marker. At the same time the shape is preserved by overshining. The new algorithm used for RAMP[10] uses a rough (and due to partly occlusion possibly slightly incorrect) pose estimation to project the circles into the image plane. When projected these circles generally become ellipses. The semi-axes are compared to the measured ones. If the anisometry of the ellipse, i.e. the ratio of the length of the major semi-axis to the minor semi-axis, differs by more than a certain threshold the fiducial is detected as partially occluded.

7.5.3 Results

We experienced an algorithm that does not falsely identify partial occlusion. To show the sensitivity of this way of detecting partial occlusion in RAMP[10] a sequence of images of a static scene has been recorded. In that sequence one fiducial is occluded by a moving finger. Rotation and translation matrices are extracted from each image of the sequence with and without the presented algorithm for detection of partial occlusion. Figures 35 and 36 show the deviation from the average values. Two conclusions can be drawn from these figures. Firstly translation is not influenced much by wrong measurements from one partially occluded fiducial. This is because the eight other fiducials average this error out. Secondly and in contrast to this rotation is affected a lot by these wrong measurements. With detection of partial occlusion measurements are much better. This leads to much less variation in the diagrams

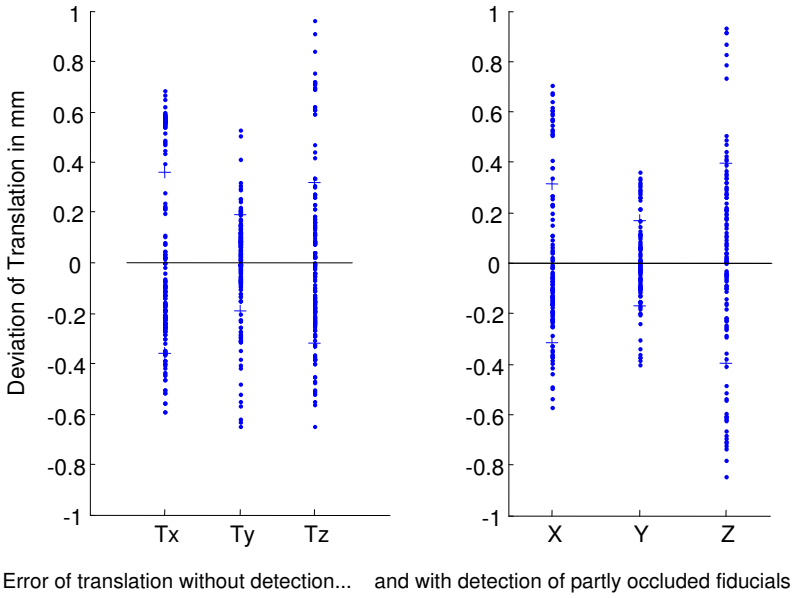

Figure 35: Illustration of deviation of translation with and without detection of partial occlusion

especially in rotation. To illustrate the impact of the error on the system: Without detection of partial occlusion 15.66% of frames would be exempted from augmentation because the error level is too high. Of course, the higher error in augmentation of the remaining frames is not desirable either. With the detection 0% of frames in exactly the same sequence were dropped from augmention. Now that we know that the detection works we would like to know how sensitive it is. The algorithm responds quite sensitively to partial occlusion but in certain cases its sensitivity decreases. These effects have been examined by trying out all configurations with simulated data. The effects are dependent on

- the angle between camera view and the plane of the circle
- the size of the occluded part of the circle
- the angle between the larger of the semi-axes and the line of occlusion

For each measurement in the diagrams, an image has been generated of the size of 50×50 pixel. An ellipse is inserted in the center of each image with a major axis of 21 pixels and an anisometry[1] of $0.6, 0.7, 0.8, 0.9, 1$. The anisometries between 0.6 and 1.0 have been chosen because this is the range of anisometries we expect in an image. The expected corresponding anisometry

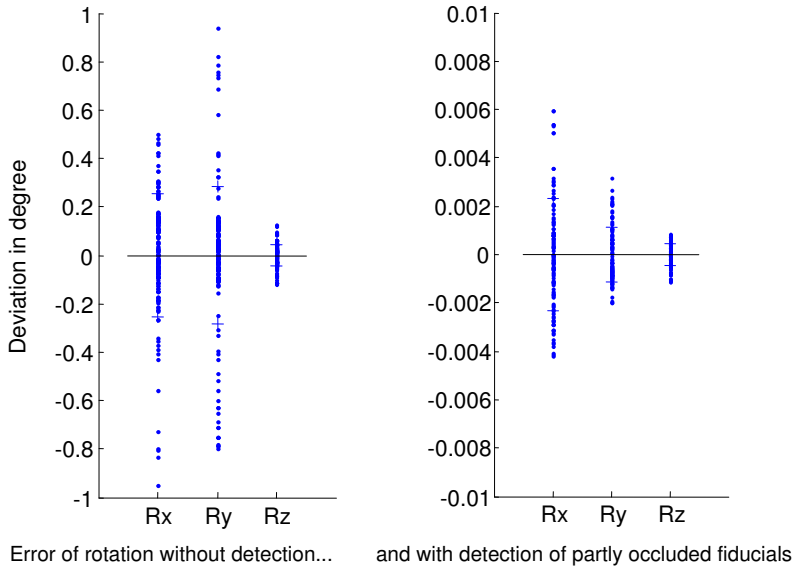

Figure 36: Illustration of deviation of rotation with and without detection of partial occlusion. Note the 100-fold difference in scale

can be obtained from figure 37 given the angle between the camera view and the circle. The smallest angles can be expected between 30° and 40°, because RAMP's[10] retro-reflective fiducials have these angles as a limit for sufficient reflection. Hence, lower angles need not be inspected. Each ellipse is turned in the image in 30 steps over 180° and the top part of each image has been occluded by setting pixels to black. The size of the occlusion has been divided into 30 steps as well. The whole set of images can be seen in figure 38. The anisometry of each resulting image has been obtained and it has been divided by the expected anisometry and one subtracted from it. This value tells us how much the measured anisometry differs from the expected one. If it has e.g. a value of 0.2 the anisometry differs by 20% and it indicates that this fiducial is partial occluded. A difference of 20% cannot occur by error in measurement. The interpretation of these figures is easy. The vertical axis in each diagram indicates the relative difference of anisometry[1]. This difference can be used for detecting partial occlusion. If the camera view is perpendicular or close to perpendicular to a circular fiducial, comparison of anisometry provides robust data for detecting partial occlusion. Unfortunately, it is possible to occlude a fiducial partially without changing anisometry if the view on the fiducial is not orthogonal to the camera view. This phenomenon is limited to a certain relative size of occlusion. This limitation depends on the

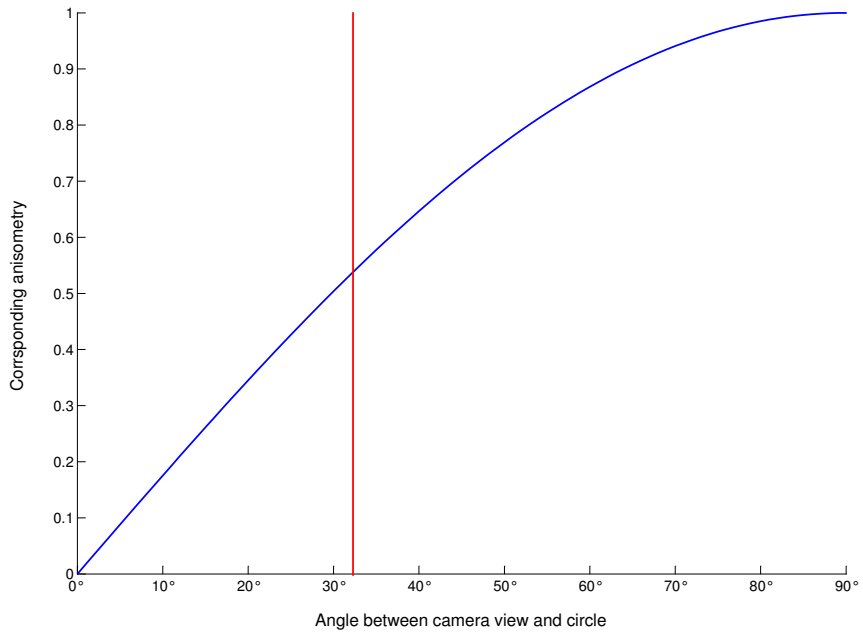

Figure 37: This diagram shows the expected anisometry of an ellipse in an image for a certain angle between camera view and its corresponding circle. The red line indicates a rough limit of reflection. Below $30° - 40°$ fiducials reflect too little light, so in RAMP, such angles may result in no ellipse at all. The function is given in equation 51 and its geometry can be seen in figure 52

angle between camera view and circle. If the relative size of the occlusion is large enough all partial occlusions can be determined robustly. This data matches our prior observation.

7.5.4 Discussion

To summarize, there is good news and bad news. As good news there is the fact that for angles over 60° the implemented algorithm for detecting partial occlusions is robust. The bad news is that for angles as low as 45° only large partial occlusion will be detected certainly, while small occlusions will be detected only if a certain side the fiducial is covered. last but not least, I would like to point out that this algorithm cannot detect partial occlusion with the same sensitivity in all situations but it improves the robustness of tracking a great deal as shown in the empirically collected data.

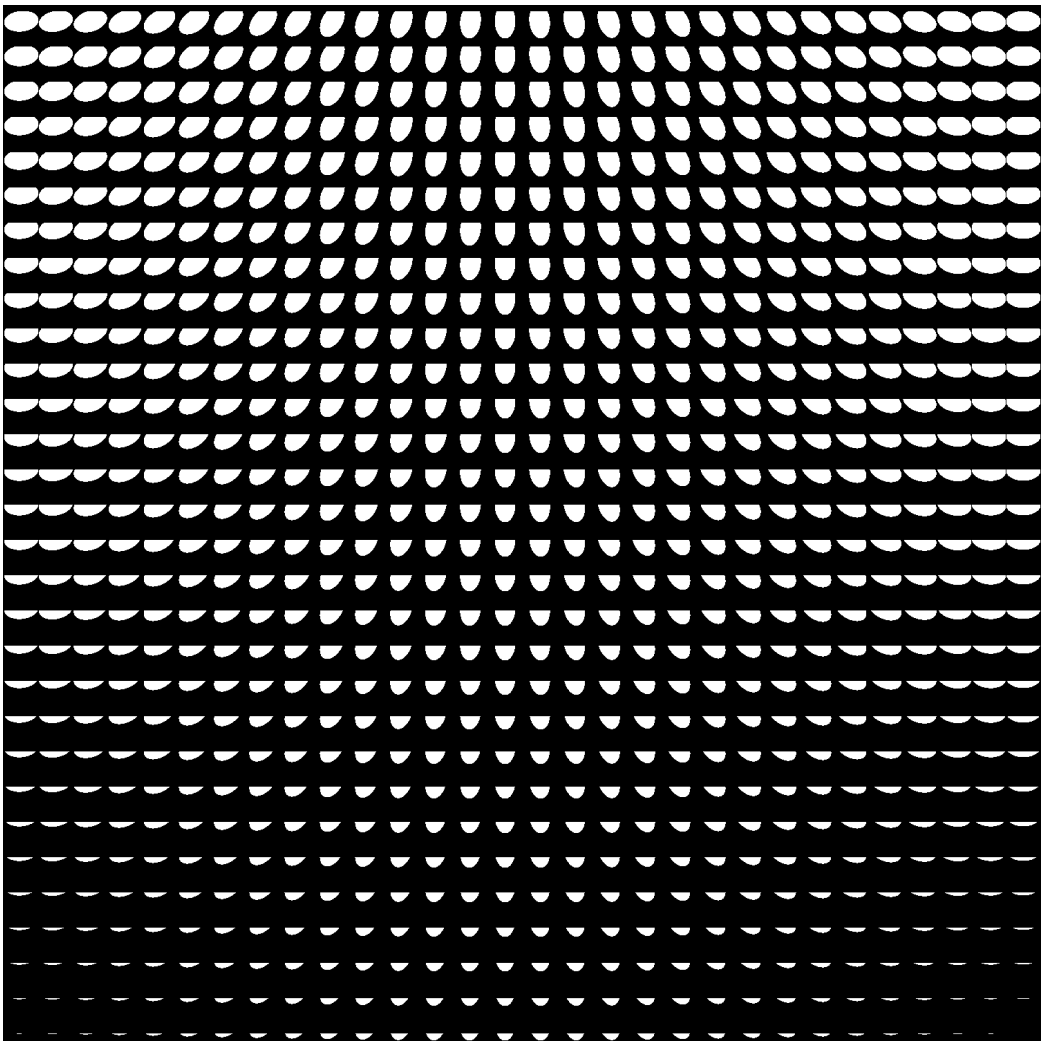

Figure 38: These artificial images have been used for generating diagram a with an anisometry of 0.6, see figure 43

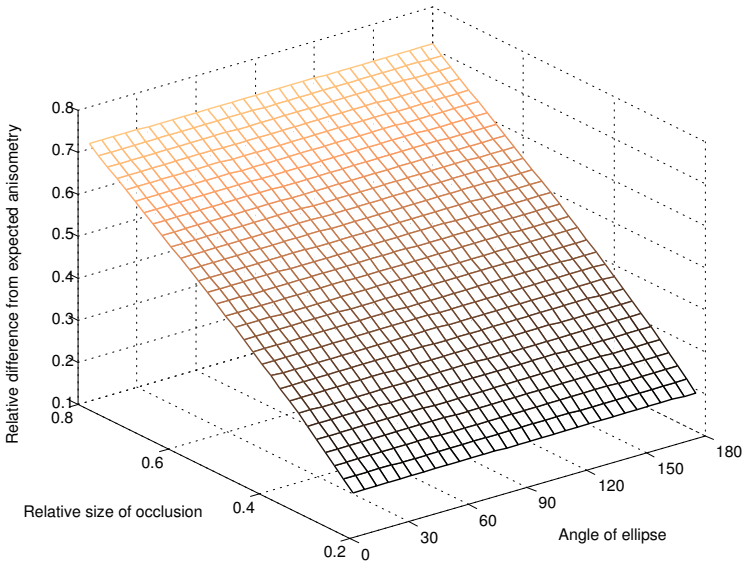

Figure 39: This diagram shows the possible sensitivity of the algorithm to detect half occlusion if the fiducial in the image has an anisometry of 1.0. The higher the value of relative difference on the vertical axis the better. In the current system a fiducial is marked to be partly occluded if its value is 0.15. An anisometry of 1.0 means that the view is perpendicular to the circle

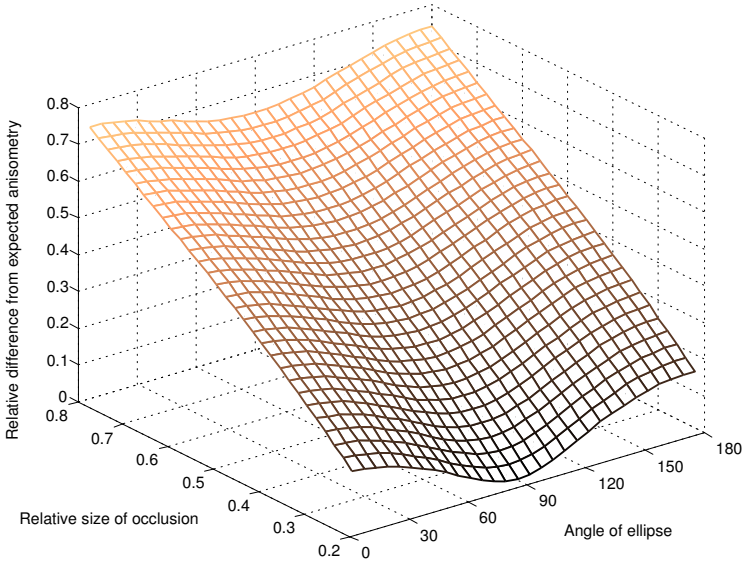

Figure 40: Similar to figure 39, but fiducials have an anisometry of 0.9. This corresponds to about 65 degrees as can be gathered from figure 37.

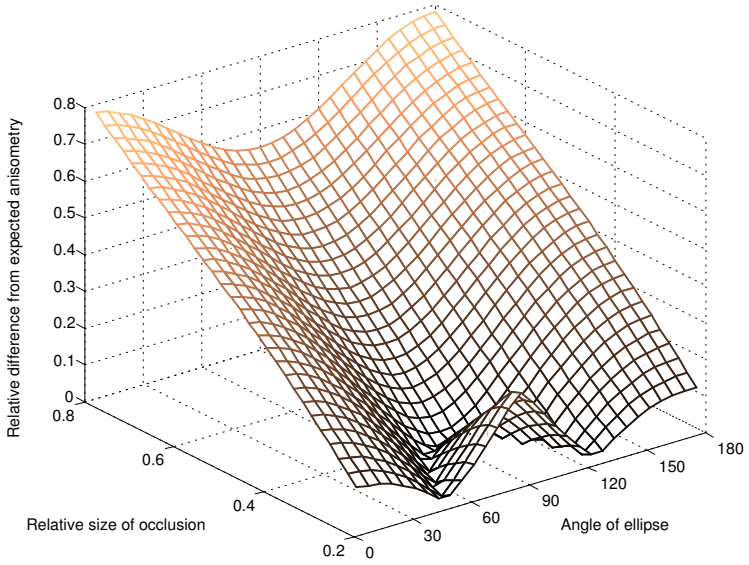

Figure 41: Anisometry of 0.8. This corresponds to an angle to the camera view of about 55 degrees as can be gathered from figure 37.

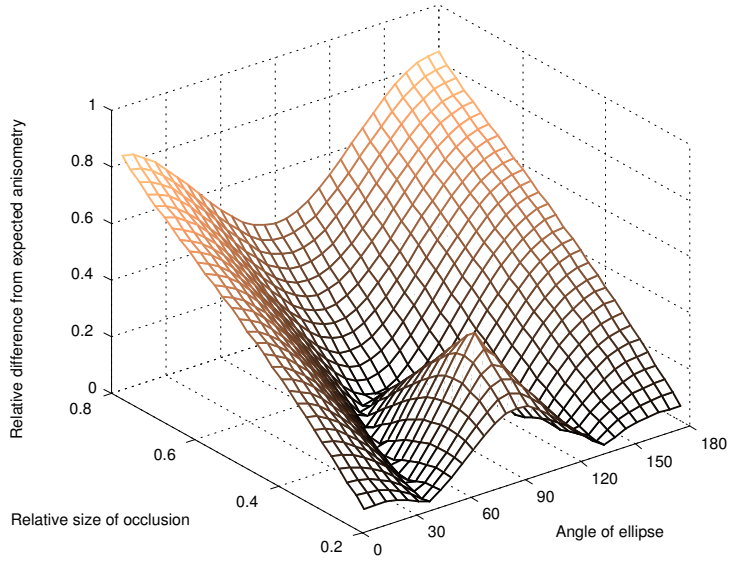

Figure 42: Anisometry of 0.7. This corresponds to an angle to the camera view of about 45 degrees as can be gathered from figure 37.

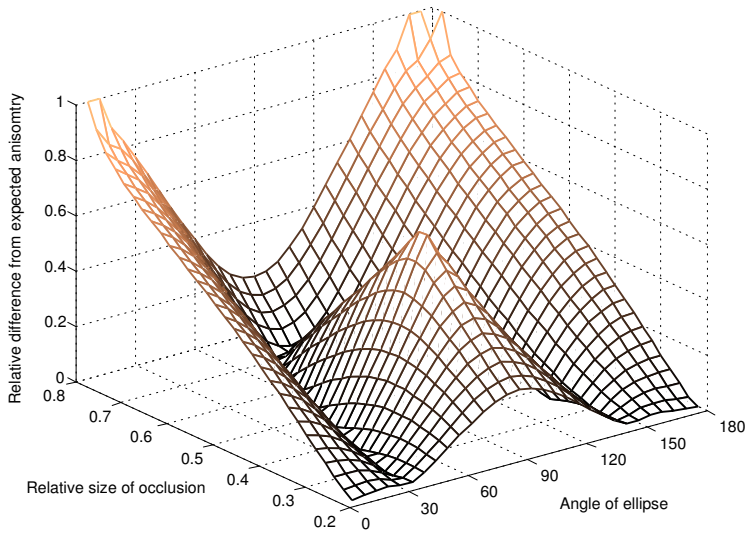

Figure 43: Anisometry of 0.6. This corresponds to an angle to the camera view of about 35 degrees as can be gathered from figure 37.

7.6 Pose estimation of single fiducials

7.6.1 Description

It is an interesting question whether there is more information hidden in the image than is used up to now. If we can use more information we might reduce the number of fiducials at the same quality of measurements or we might make the tracking more versatile or more robust. In this case we make use of the semi-axes of fiducials. The detection of partial occlusion only makes use of the ratio between major and minor semi-axes, but the absolute size of them is discarded. Since the size of the semi-axes can be determined precisely in RAMP[10] we would like to estimate the distance to the camera of the fiducial in order to obtain 3D information about the fiducials. Gaining 3D information is supposed to support marker identification in the next subsection but since it yields more possibilities than just this, it is discussed in its own subsection.

7.6.2 Approach: Calculating the distance to the camera based on the larger of the semi-axes

There is a simple relationship between the distance between the camera and the circle, and the bigger semi-axis of an ellipse in an undistorted image. This means that we can obtain the distance of a circular fiducial from the major semi-axis. The equation is $d = \frac{r}{a} f$ in which d denotes the distance, r denotes the radius of the circle, f denotes the focal length and a denotes the length of the major semi-axis in the image on an undistorted sensor. This relationship is proven in the appendix (see section 9.2.5).

In order to obtain a fast computation of the semi-axes in undistorted image coordinates we use an approximation, so we need not undistort each pixel that is included in the fiducial in the image. As an approximation the distortion preserves the centroid and the length of the semi-axes. Thus, we compute the full axes of the ellipse in distorted image coordinates and undistort only the coordinates of the ends of the full axes and the centroid. By this means we obtain an approximation to the undistorted major axis.

7.6.3 Results

Due to the effect of overshining (see figure 34) the semi-axes can appear bigger than they are. On the other hand fiducials appear smaller than they are if the exposure is too little. By brightening or darkening the flash we can change the size of fiducials in the image considerably. This leads, of course, to a wrong estimation of depth.

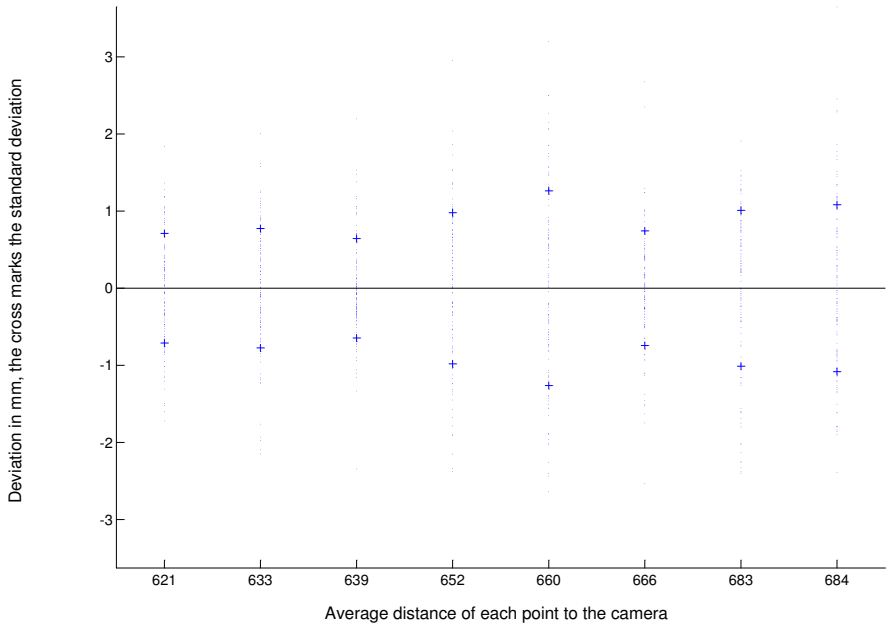

Figure 44: These are the results of the experiment with eight static markers in 118 images. The distance between markers and camera $621 - 684mm$ which is within the range of distance RAMP is designed for.

Nevertheless it is an interesting question how sensitive to noise this approach to depth estimation is. Eight static markers have been observed in a sequence of more than one hundred images in order to find out how precise these values are. Comparison between one and the same point in different images of a sequence can reveal the amount of jitter that is due to noise. Since the light and the positions of camera and fiducials do not change in the image sequence, the error that is due to noise of the camera. Figure 44 shows the result of the experiment. The highest absolute deviation in this experiment is lower than $4mm$ or 0.642% and the average standard deviation has been as low as $0.901mm$ or 0.144%.

7.6.4 Discussion

The precision of the results was positively surprising even though this way of exterior camera calibration is worse than solving a perspective eight-point problem by optimization. After an optimization of all eight markers the precision in depth is, of course, much higher. As a result of an optimization

of the same image sequence the standard deviation has been $0.05mm$ or 0.00803%. The remarkable fact is that we can deduce a millimeter-precise estimation of each fiducial just from knowing its diameter.

As mentioned above these values are precise, but not accurate. Since we know that this kind of pose estimation is precise we know that it might be worth taking measures to maintain accuracy as well. As a suggestion ring-shaped fiducials can be evaluated to estimate the middle between the inner and outer radius. They can also be measured by moments, so the results have the same precision. These fiducials can be produced as easily and accurately by simply punching out. As an advantage the inner and the outer radius suffer from overshining in the same way. Therefore the middle of it is expected not to be affected by overshining.

A better approximation for undistorting semi-axes can be obtained by undistorting five arbitrary points on the ellipse and take the semi-axes of the ellipse given by these five points.

7.7 Fiducial identification

7.7.1 Description

Before solving perspective n-point problems for exterior camera calibration all of the fiducials extracted from the image have to be matched to their counterparts in the model. This is called identification. Simply trying every possibility takes too much time as explained in 4.8. In RAMP[10] it used to be done by algorithms specially designed for each set of fiducials (see section 6.4.3). Since each fiducial has the same appearance, each one has to be identified by geometric cues. We know that each fiducial in a set of fiducials is attached to a rigid body. Hence, markers have fixed distances in 3D. Unfortunately, the distance in 3D is not preserved by the projection to 2D. The wanted algorithm has strong requirements.

- **Real time computation**. The algorithm must solve the problem in a time window of less than five milliseconds on our machine.

- **Missing markers**. The algorithm must recognize the set of fiducials even if one or more fiducials are missing e.g. due to occlusion. *Any* fiducial might be occluded.

- **Additional markers**. The algorithm must not be confused by other markers of other sets of fiducials. Additional markers might appear in the vicinity of the correct markers as well. Note that for algorithm design this problem is quite different from than the one above.

- **No restrictions in fiducial set design**. The algorithm must handle different kinds of sets. Furthermore we do not want any restrictions about the side from which we approach the set of fiducials e.g. "Fiducial one and two are always at the top of the image".

In other words: We cannot rely on the visibility of any of the fiducials. We do not know if a fiducial belongs to the set the algorithm looks for and we do not know if a certain fiducial of the set is in the image. This uncertainty causes many combinations of possible situations that make it difficult to design an algorithm for the real time constraint.

Since the algorithm cannot rely on single fiducials, it has to find a solution that is the best one for the given image and model. To find out about the quality of three (or more) identified fiducials, three points are taken to calculate up to four transformation matrices which are the solutions to the perspective three point problem (see 4.7). All of the points in the 3D model are projected into image coordinates according to each of the four transformation matrices. The combination of matches of fiducials and their average distance between projected points and measured fiducials is taken as a norm to find out the best solution among the four matrices. A match is found if the distance between a projected and an actual fiducial is below a certain threshold. The norm of the best solution also reveals the quality of that identification compared to other hypothetical identifications. Fiducials that are not identified yet can be assigned if they are close enough to the projected model point. This way of identification is very robust if the first three fiducials have been identified correctly. Unfortunately, each try is computationally expensive. To give an impression of proportions: It takes the current system about $220 - 250\mu s$ to calculate the described step of estimating the transformation by three points and model projection in comparison to $5000\mu s$ for the whole identification procedure. It is clear that we have to collect more knowledge to limit the number of solutions to examine.

7.7.2 Approach: Using pose estimation of single fiducials for 3D-3D registration

As explained in section 7.6 we can estimate the 3D position of a single fiducial precisely at little computational costs. A 3D-3D registration yields faster computation than 2D-3D registration in the case of three points in principle because the solution is not ambiguous and need not be computed by iteration. Most 3D-3D algorithms (as in [31]) target large sets of points that try to assign most of the points correctly e.g. ICT. ICT assigns the points randomly, swaps pairs of points and keeps changes if the solution gets better. This

algorithm can run into local minima, because only two points are swapped at a time. Unfortunately, we need all of the points to be assigned correctly otherwise the solution of the perspective n-point problem is not correct at all.

The presented algorithm can rely on the fact that 3D distances remain the same after projection, so neighboring points remain neighboring points after projection. This is only true for a 3D-3D projection. The algorithm looks for the two nearest neighbors of each fiducial in the image. A coordinate system is generated with these three points and it is scaled to the distance between the point and its first neighbor. The scaling would not be necessary if the pose estimation of markers were accurate, but it is only precise (see section 7.6.3). The same is done for each marker in the model. The coordinate systems calculated for each fiducial in the image are compared with the ones in the model. From the best matches of coordinate systems the first point and its two neighbors are taken for an accurate identification as explained in the description of the problem. This is necessary in order to reduce the chance of ambiguous setups as much as possible.

This is the algorithm in pseudo code.

```
function findBestMatch(markersInModel, markersInImage, tolerance) {
  mostMatchedPoints = 0;
  lowestDistance = Inf;
  bestN=0;
  bestM=0;

  for n = 1 to markersInImage.length
    turnedMarkersInImage = changeCoordSystem(markersInImage, n);
    for m = 1 to markersInModel
      turnedMarkersInModel = changeCoordSystem(markersInModel, m);
      [matchedPoints, distance] =
        norm(turnedMarkersInModel, turnedMarkersInImage, tolerance);

      if matchedPoints > mostMatchedPoints or
      (matchedPoints == mostMatchedPoints and
       distance < lowestDistance)
          bestN = n;
          bestM = m;
      end; //if
      end; //for
    end;  //for
  return [n,m];
```

```
}

// distance between sets of points
function norm(markerSet1, markerSet2, tolerance){
  matches = 0;
  sumDistances = 0;
  for n = 1 to markerSet1.length
    for m = 1 to markerSet2.length
      distance = norm(markerSet1[n], markerSet2[m])
      if distance < tolerance
        matches = matches + 1;
        sumDistances = sumDistances + distance;
      end; // if
    end; //for
  end; //for

  return [matches, distance/matches];
}

// distance between points
function norm(marker1, marker2){
  return sqrt( (marker1.x-marker2.x)^2+(marker1.y-marker2.y)^2
              +(marker1.z-marker2.z)^2);
}

function changeCoordSystem(markerSet, center){
  base1 = markerSet[center]- markerSet[center].firstNeighbor;
  base2 = markerSet[center]- markerSet[center].secondNeighbor;

  // Distance between a marker and its first neighbor is set to 1
  base1 = base1 / abs(base1);
  base2 = base2 / abs(base2);

  //base2: The orthonormal part of the line between a point and
  //       its second neighbor
  base2 = base2 - dotProduct(base1, base2);
  base2 = base2 / abs(base2);

  //base3: Orthonormal vector to base1 and base2
  base3 = crossProduct(base1, base2);
```

```
  for n = 1 to markerSet.length
    markerSetToReturn[n].x = dotProduct(markerSet[n],base1);
    markerSetToReturn[n].y = dotProduct(markerSet[n],base2);
    markerSetToReturn[n].z = dotProduct(markerSet[n],base3);
  end; // for
}
```

The function `findBestMatch` returns an identification of one fiducial in form of a pair of numbers representing the correspondence between image and model. The first and second neighbor of each fiducial must correspond as well, if visible. With these three identifications the perspective three-point problem can be solved to identify all of the fiducials correctly. The `tolerance` is the maximum distance between a marker and a point in the model to be matched. The algorithm used in RAMP[10] is a little modified. It returns not a mere identification but a number of identifications to be tried with the more accurate method mentioned above. This has been added because of the following observation. Due to the scaling the identification that is deemed to be best one can still be the wrong. Nevertheless the correct solution is one of the best. The accurate method can clarify this situation. Through this combination we can enjoy the advantages of less precise and inaccurate 3D pose estimation of each point as well as the advantage of the accurate and slow 2D pose estimation. Note that just 2D estimation cannot find out whether a fiducial is occluded by another one, while 3D estimation can.

As another modification in RAMP[10] no point in the model can be matched twice because this would definitely produce bad results and favor odd solutions. As a further modification of the presented algorithm not all of the combinations of image points and model points are taken into account. In most cases the result of a combination and a combination with one of the neighbors is the same. For visualization of this observation refer to figure 45. The information the algorithm deduces from each combination is whether or not a fiducial and its two neighbors are part of the model. If so, a solution has been found and the algorithm could stop if it was aware of it. If this triangle of a point and its neighbors is not part of the model there are two possibilities. First, all of the fiducials belong to another set of fiducials, so the neighbors will not produce a valid match, either. Second, the neighborhood relationship between the three points of the marker set is disturbed by an additional or a missing point. If the relationship is disturbed, the chance is very little that one of the neighbors has two neighbors that belong to the marker set. In other words, if the neighborhood relationship of markers in a set is disturbed by an a missing or a surplus point, the chance is little that the neighborhood relationship of the two neighbors does not suffer from the

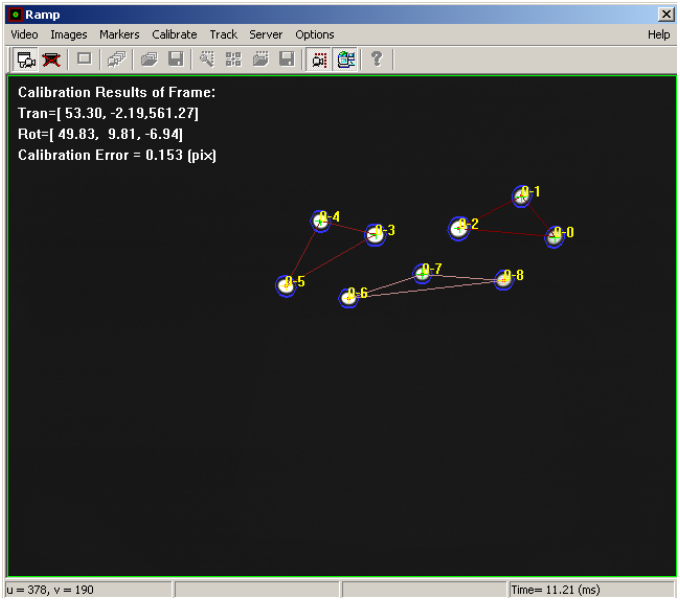

Figure 45: Due to 3D distribution fiducial 3 and 5 are nearest neighbors of fiducial 4 and equivalently, 3 and 4 are nearest neighbors of 2. The same applies for the other combination. These three combinations will gain the same information. The information we deduce is whether or not a fiducial and its two neighbors in the image is part of the model

same problem. As a guideline we cannot depend on just a chance. We can either maintain accuracy in a given situation or we cannot. In those cases the robustness of the algorithm must rely on another triplet of points. For that reason in RAMP[10] the algorithm has been extended by a heuristic not considering fiducials that have already been used as neighbors in a comparison. Note that they may be used as neighbors of other markers but are not considered for finding their own neighbors and producing a coordinate system for comparison.

These modifications do not alter the complexity but they disturb the readability of the pseudo code, and are thus not involved in the code above. Why has this heuristic been added although it does not change the complexity of the algorithm? The reason is not the speed up of only about 1:3. The answer is easy but not obvious. The algorithm not only takes about a third of the time, which could be more than undone by adding a single fiducial in the image. It is interesting to note that is that without the heuristic the algorithm yields about three times as many redundant solutions to be evaluated with the slow back projection method. As stated in the description (7.7) reducing the number of possible solutions is the aim of the whole procedure.

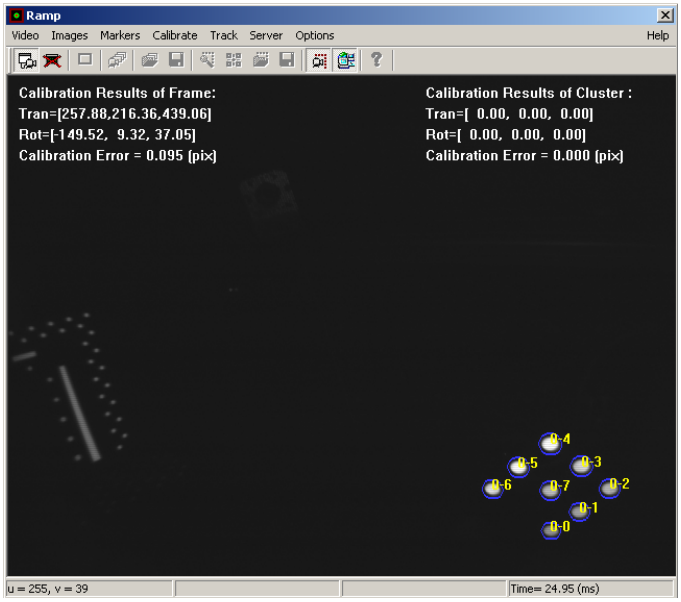

Figure 46: The marker set is in the lower right hand corner. Blue dots represent the boundaries of of an extracted point. The time depicted on the bottom of the picture is the time for exterior camera calibration plus visualization of one image.

7.7.3 Results

Looking at the pseudo code it is easy to see that the algorithm comes up with $n \cdot m$ possible matches. n denotes the number of fiducials in the image and m denotes the number of points in the model. In the current implementation every comparison takes $O(n \cdot m)$ computations because each point in the model is compared to each one in the image to find the nearest match. By this means the number of possible solutions is limited to $n \cdot m$ solutions compared to $n!$ possible solutions when employing a brute force method. These solutions can be compared directly without an estimation of the transformation matrices and without evaluating matrix multiplications for back projection. This explains why the algorithm is fast enough to meet the time constraints without cutting down the other constraints.

The algorithm has been tested in RAMP[10] for use without failure. Like the other new features it has even proven its robustness in a clinical trial of the whole system. Of course, this versatile algorithm has its limits. If physical arrangement of the fiducials is ambiguous, i.e. two or more different positions of the sets of fiducials can result in exactly the same image, no algorithm can produce a clear result. All sets of fiducials that are symmetric in rotation

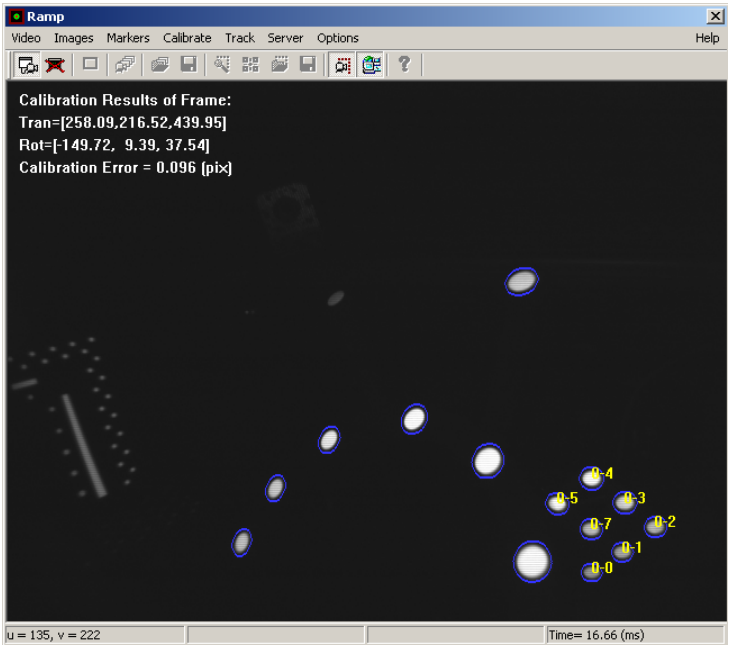

Figure 47: The same setting as in figure 46 with an additional set of fiducials occluding one point

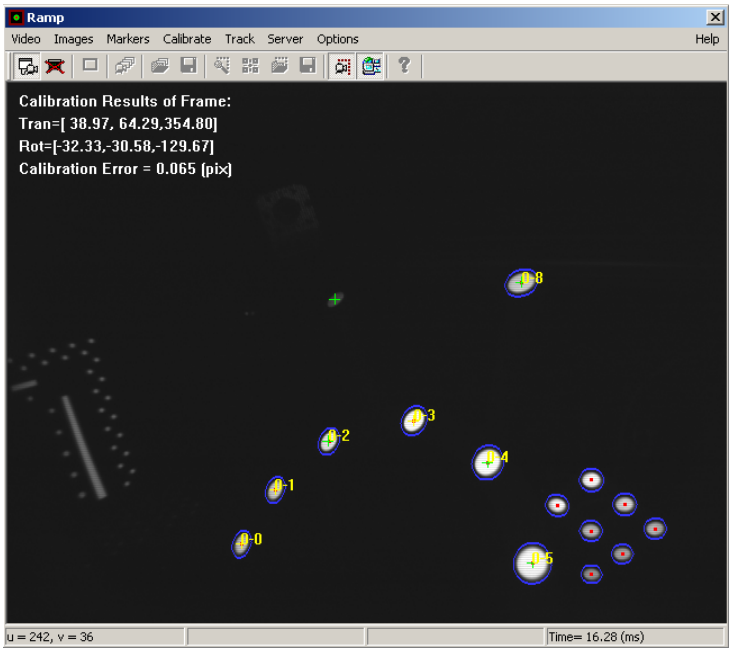

Figure 48: The same sets of fiducials as in 47 but the algorithm searches for the other set. Red points indicate the center of found fiducials and green crosses indicate where the algorithm expects to find a fiducial

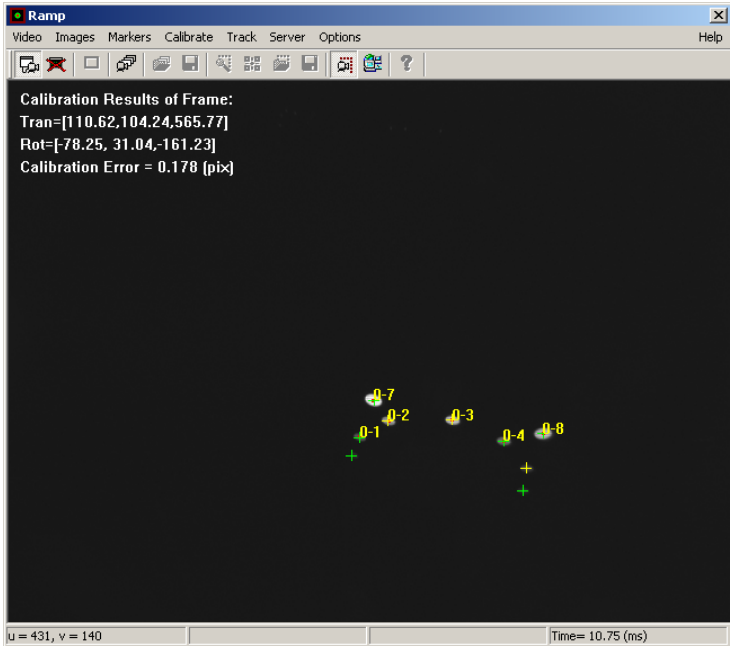

Figure 49: The same set of fiducials as in figure 48. This demonstrates that the algorithm can handle difficult positions as well

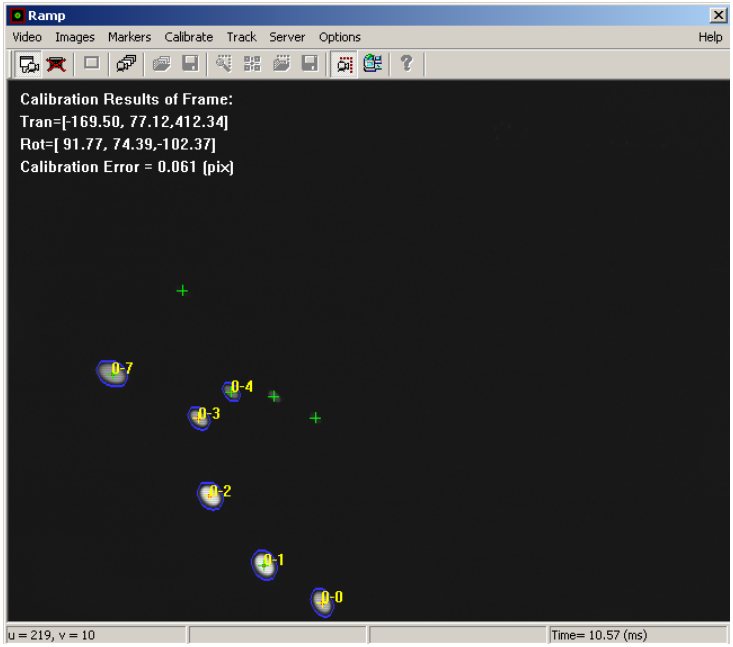

Figure 50: The same set of fiducials as in figure 49

produces ambiguous images. An example for a bad set of fiducials is a circle of equidistant points. Some ambiguities are not as obvious as symmetry. They simply appear when some fiducials are occluded. As an example take a half circle of equidistant fiducials. If all of the fiducials are present there is no ambiguity. If one point at one of the ends is missing, it cannot be decided anymore from which end the point has been removed unless further knowledge is available.

There is a restriction to the algorithm about the fiducial set design. The algorithm works most robustly if the nearest and the second nearest neighbor of each fiducial are not ambiguous, but this is not a strong restriction.

Note that the algorithm can solve ambiguities that only appear after projection but not in three dimensions. As an example, take a circle of equidistant points where at least one point is lifted from the plane of the circle. Taking an image perpendicular to the circle, all of the points appear to be equidistant. The algorithm takes 3D information into account and hence, it is able to identify all of the points correctly. This behavior makes a marker set design easy and straightforward. More important, it saves us from surprising results due to ambiguities hidden by perspective projection.

7.7.4 Discussion

The presented algorithm to find parts of a certain set of fiducials in an image is fast enough, versatile and robust. This algorithm enables RAMP[10] to do easy research and implementation of ergonomic fiducial set designs for special tasks. Now, the design can follow its function and without restrictions of the algorithm for identification.

7.8 Multiple marker set separation

7.8.1 Description

After the set of fiducials that provides the world coordinate system is found, there might be other sets of fiducials in the image. These sets are attached to a tool RAMP[10] is supposed to track, too. To employ the same algorithm as in section 7.7 subsequently would be a possibility to solve the problem. If many sets of fiducials are in the image the optimization step for estimating transformation matrices for each set of fiducials takes up a significant amount of of time. Therefore the time left for identification is continuously decreased.

7.8.2 Approach: Ring-shaped fiducials instead of solid ones mark the center of an independent set of fiducials

As a trade-off we violate our paradigm about equal looking fiducials. Ring-shaped fiducials are introduced into the set. The sets of fiducials for tools have a specific shape (see figure 21 set number 2). The fiducial in the center of the set has been changed to a ring-shaped fiducial. Since the remaining fiducials in the image belong to sets of fiducials of tools that have this specific shape all of the fiducials are assigned to the set with the next ringed fiducial.

7.8.3 Results

This technique of associating fiducials with their set is very fast compared to the one presented in section 7.7. Finding out whether a fiducial is a ring takes $O(n)$ computation with n being the number of fiducials. This is because for each point the intensity of the centroid has to be compared to the average intensity of the marker. The intensity of a fiducial is calculated for the moments anyway, so the algorithm gets it for free. To find out for each fiducial which ring-shaped fiducial is the next one has the complexity of $O(n \cdot s)$ where s denotes the number of sets.

As a short result, this way of associating fiducials with their sets has been tested in the system for two tools at the same time in the image. If more than two tools enter the scene it is too narrow for all of the tools to be tracked in the image. The attached sets of fiducials must be visible at all times in order to be tracked. The sets of fiducials must cover a certain space to maintain the desired accuracy as explained in 5.5. According to the clinical test two tools at the same time already restrict the freedom of the tools too much for an impression of undisturbed work.

Misclassifications have only been observed when the sets occluded one another. In this case augmentation is stopped anyway because the size of sets is at its lower limit, so pose estimation with fewer points might be too inaccurate.

7.8.4 Discussion

This way of distinguishing fiducials is fast and in our tests the classification has been correct. Further theoretical investigation of the problem is not planned because the problem of a scene too narrow for more than one tracked tool has to be attacked first of all.

8 A look into the future

The way of tracking described in this Diplomarbeit yields high accuracy measurements in real time for a reasonable price. The new algorithms could make the system more robust and versatile. Especially with the new algorithm for enumeration single camera tracking does not suffer from the problem of identification of fiducials anymore. So it can be used without limits to the design of the sets of fiducials. Since only one camera view is necessary, fiducials should be less likely occluded than in multiple view systems.

There are still open questions for a more convenient tracking. I have presented a way of depth estimation of single circular fiducials. The effect of overshining distorts the results. Can we get accurate estimates with shapes other than circles like rings for instance? If we obtained accurate results by depth estimation could we detect partial occlusion more robustly?

This way of tracking can track multiple sets of fiducials limited by the size of the image. For three different sets the image is already too crowded for convenient work. How can we reduce the size of the sets of fiducials without losing too much accuracy? Is there an optimum spatial distribution for a set of fiducials?

9 Appendices

9.1 Semi-axes extraction of ellipses from second order moment of its area

As future work to be done there is subpixel-accurate estimation of rings as opposed to circles. Therefore I would like to present the theory leading to the formula how semi-axes of solid circles can be obtained from the moments of its region presented in section 5.3.

Any ellipse centered in (0,0) and semi-axes of (a,0) and (0,b) follows the ellipse equation:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \tag{33}$$

In matrix notation as an quadratic form

$$\begin{pmatrix} x \\ y \end{pmatrix}^T \begin{pmatrix} 1/a^2 & 0 \\ 0 & 1/b^2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 1 \tag{34}$$

The matrix of the quadratic form for oblique ellipses can be described [35] as

$$V \begin{pmatrix} 1/a^2 & 0 \\ 0 & 1/b^2 \end{pmatrix} V^{-1} \tag{35}$$

with a, b as lengths of semi-axes. The semi-axes are contained in V normalized to the length 1. Obtaining these from an arbitrary matrix results in having to solve an eigenvector problem. As another interpretation of the matrix V, it performs a rotation that leads to the obliqueness (see section 4.6.3).

Calculating the second order moment of an ellipse as described in equation 34 results in

$$C = \frac{1}{A} \begin{pmatrix} \iint x^2 dxdy & 0 \\ 0 & \iint y^2 dxdy \end{pmatrix} \tag{36}$$

Evaluating the integrals we get (for detailed calculations see 9.1.1)

$$C = \frac{1}{A} \begin{pmatrix} \frac{a^3 b}{4}\pi & 0 \\ 0 & \frac{ab^3}{4}\pi \end{pmatrix} \tag{37}$$

With the area

$$A = \iint 1 dxdy = ab\pi, \tag{38}$$

of any ellipse with a and b as semi-axes we get

$$a = 2\sqrt{\lambda_1}, \quad b = 2\sqrt{\lambda_2} \tag{39}$$

A prosaic interpretation of these formulas is the following. If we interpret the image as an stochastic experiment we can determine the length of the semi-axes from the variances of point in the direction of each semi-axes. Now any point in the ellipse is designated as 1 and the others are designated as 0. The expected value of this experiment is the center of the dot. A principle component analysis of measurements yields the eigenvectors of the covariance matrix which tell us the direction of the semi-axes. The exact length of the semi-axes can be calculated from the variation and the size of the area.

9.1.1 Evaluation of integrals

To show:

$$C = \begin{pmatrix} \int\int x^2 dxdy & 0 \\ 0 & \int\int y^2 dxdy \end{pmatrix} = \begin{pmatrix} \frac{a^3 b}{4}\pi & 0 \\ 0 & \frac{ab^3}{4}\pi \end{pmatrix} \tag{40}$$

Using the ellipse equation 33 to generate the boundaries we get

$$C_{11} = \int_{-a}^{a} \int_{-b\sqrt{1-\frac{y^2}{a^2}}}^{b\sqrt{1-\frac{y^2}{a^2}}} x^2 dydx \tag{41}$$

Evaluation of one integral results in

$$C_{11} = 2b \int_{-a}^{a} x^2 \sqrt{1 - \frac{x^2}{a^2}} dx \tag{42}$$

Substitution of x

$$x := a \sin t \tag{43}$$

$$C_{11} = 2b \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} a^2 \sin^2 \cos ta \cos t dt \tag{44}$$

$$C_{11} = 2ba^3 \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \sin^2 \cos^2 t dt \tag{45}$$

Substitution

$$\sin^2 t \cos^2 t = \left(\frac{\sin 2t}{2}\right)^2 \tag{46}$$

$$\sin^2 t = \frac{1 - \cos 2t}{2} \tag{47}$$

$$C_{11} = \frac{2ba^3}{8} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} (1 - \cos 4t)\, dt = \frac{ba^3}{4}\pi \tag{48}$$

The calculation of $C_{22} = \int\int y^2 dxdy = \frac{ab^3}{4}\pi$ is similar.

9.2 Proof of formula used in section 7.6

In section 7.6 a formula is used to determine the distance of a single circular fiducial to the camera by the length of the major semi-axis in the projected image.

First of all, in section (9.2.1) it will be shown that if a line is projected to a coplanar plane, we can obtain a simple relationship by similar triangles. A line that is coplanar to a plane is perpendicular to its norm which is referred to as 'the view of the camera' in this proof. Therefore it is proven in section 9.2.3 that the projected diameter of a circle is the same as the major axis of the resulting ellipse or in other words that the longest projection of all diameters of a circle must be a perpendicular one. Finally, theorem 9.2.5 concludes the desired equation used in section 7.6 and a prosaic geometric explanation is given.

9.2.1 Proposition

Given two points P_0, P and a pin hole projection described by the matrix M and the focal point F, the following rule applies:
If

$$(P - P_0) \perp (F - P_0) \wedge (MP - MP_0) \perp (F - MP_0)$$

it follows

$$\frac{|P - P_0|}{|MP - MP_0|} = \frac{|F - P_0|}{|F - MP_0|}$$

9.2.2 Proof

The proposition is proven by the similar triangles of $\triangle(P, P_0, F)$ and $\triangle(MP, MP_0, F)$

9.2.3 Proposition

Define the line l as the line between point P and the circle center P_0. Let F be the focal point of a pinhole projection. The projection of l is longest given a certain distance $d = P_0 - F$, if and only if

$$F - P_0 \perp P - P_0$$

for $d > r\sqrt{2}$

9.2.4 Proof

See figure 52

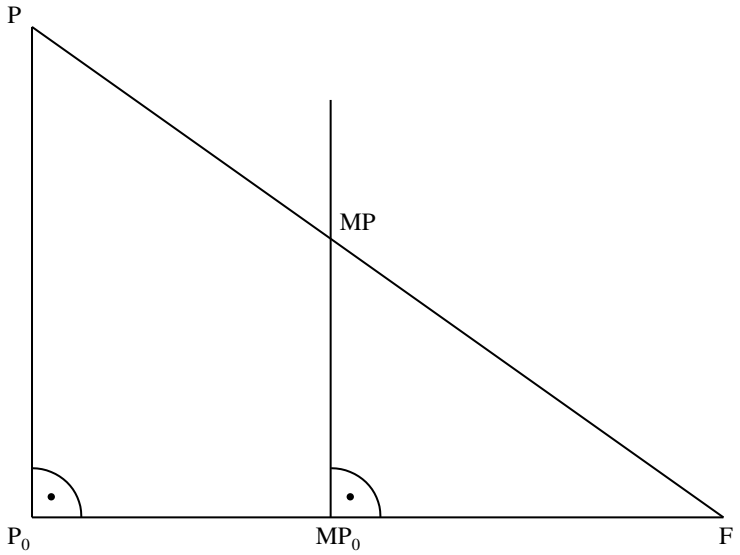

Figure 51: Geometry of Proposition 9.2.1

$$I_1 = \frac{dr\sin\alpha}{d - r\cos\alpha}, \quad I_2 = \frac{dr\sin\alpha}{d + r\cos\alpha} \tag{49}$$

$$i = \frac{f}{d}\left(I_1 + I_2\right) \tag{50}$$

$$i = \frac{fr\sin\alpha}{d + r\cos\alpha} + \frac{fr\sin\alpha}{d - r\cos\alpha} = \frac{2dfr\sin\alpha}{d^2 - r^2\cos^2\alpha} \tag{51}$$

$$i'(\alpha) = \frac{(2dfr\cos\alpha)(d^2 - r^2 - r^2\sin^2\alpha)}{d^2 - r^2\cos^2\alpha} \tag{52}$$

$$\Rightarrow i'(\alpha) > 0 \quad \text{, if } d > r\sqrt{2} \tag{53}$$

$$\Rightarrow argmax_{\alpha\in[0^\circ,90^\circ]} i(\alpha) = 90^\circ \quad , if\ d > r\sqrt{2} \tag{54}$$

$$(53) \wedge (54) \Rightarrow \left(a = \max_{\alpha\in[0^\circ,90^\circ]} i(\alpha) \Leftrightarrow P - P_0 \perp F - P_0\right) \tag{55}$$

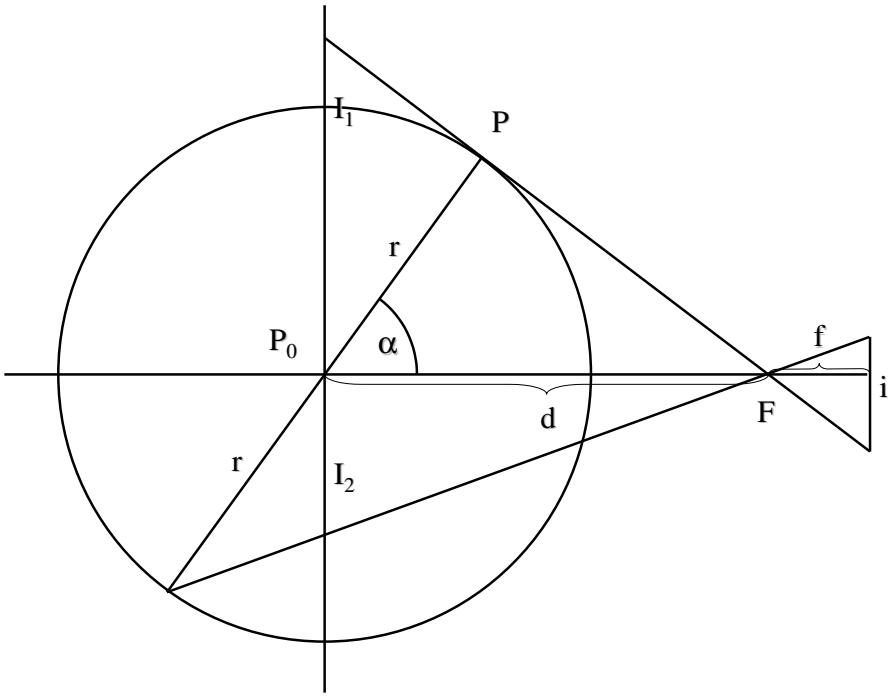

Figure 52: Geometry of proposition 9.2.3,
d denotes the distance from the circle center to the focal point,
r denotes the radius of the circle,
i denotes the diameter of the circle projected,
F is the focal point,
f denotes the focal length α denotes the angle $P - P_c$ and $F - P_c$

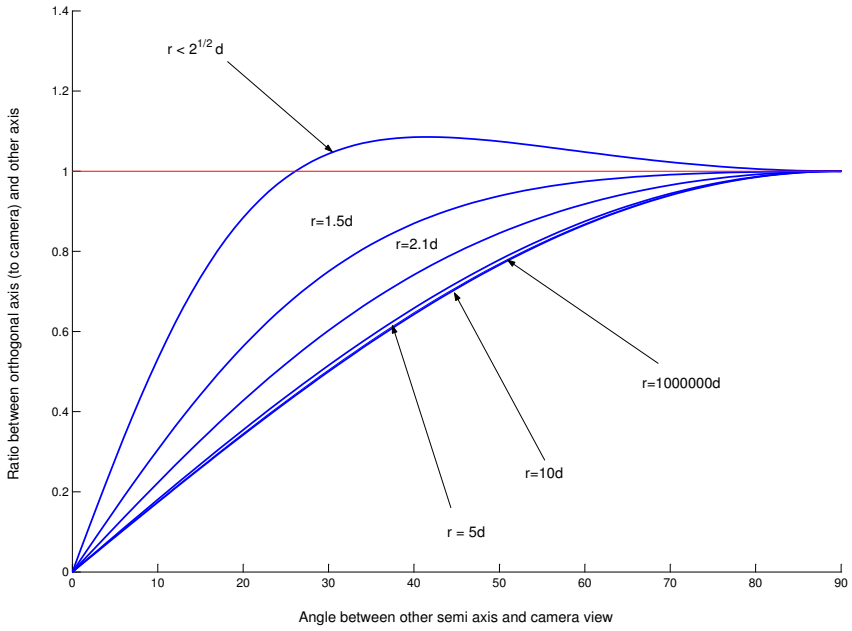

Figure 53: $i = \frac{2dfr\sin\alpha}{d^2 - r^2\cos^2\alpha}$ with $f = 1$

9.2.5 Proposition

Let us define a pinhole projection by the matrix M, its focal point F and focal length f as well as a circle C, its radius r, its center P_0. Let us define the semi-axis a, b $(a \geq b)$ of C projected by M. The distance $d = |P_0 - F|$ can be obtained easily by the equation

$$d = \frac{r}{a} f$$

for $r < \frac{d}{\sqrt{2}}$

9.2.6 Proof

Let us define $\mathbf{v} := F - P_0$
Let us define S_{circ} as the plane that holds the circle.
Let us define S_{cam} as the plane that holds the circle center P_0 and that is orthogonal to the vector $\mathbf{v}$.

1. $\mathbf{v} \perp S_{circ}$.
   This means $P - P_0 \perp F - P_0 \forall P \in C$. Thus $a = b = |MP - MP_0| \forall P \in C$ and proposition 9.2.1 leads to the desired result.

2. **otherwise**.
   Let us define L as the intersecting line of S_{cam} and S_{circ}.
   Let us define the intersections of L and the ellipse P_{e_1} and P_{e_2}.
   $(P_{e_{1/2}} - P_c \perp F - P_c) \Rightarrow^{Prop 9.2.3} a = |P_{e_1} - P_{e_2}|$ is longest diameter of the ellipse
   $\Rightarrow \frac{a}{2}$ is the bigger semi-axis of the ellipse
   This means $P_{e_1} - P_0 \perp F - P_0$ (or likewise $P_{e_2} - P_0 \perp F - P_0$) and proposition 9.2.1 leads to the desired result.

9.2.7 Explanation

Any rotation in space can be described through a vector and a rotation of the angle θ around it. Any rotation of a circle in space can be described through a vector in the plane of the circle and a rotation around it, because we can neglect the part of the rotation which axis is perpendicular to the circle.

Therefore the rotation from the circle in S_{cam} to the circle in S_{circ} can be expressed as a rotation around the line L as an axis. The length of the projection of L will only be dependent on the distance and not on the rotation. The length of a line after a projection given a certain distance to the rotation center is longest if the line is perpendicular to the line connecting

the rotation center and the focal center. This is proven in proposition 9.2.3. Therefore the bigger semi-axis corresponds to L which is not distorted by the rotation. With this information we can calculate the distance of the circle to the camera easily by using similar triangles (see 9.2.1).

10 Glossary

(1) **Anisometry**
Anisometry is the ratio between the length of the major semi-axis of an ellipse to the length of the minor semi-axis.

(2) **CCA**
Connected Component Analysis. Basic segmentation that divides an image in components connected by a neighborhood relationship.

(3) **CT**
Computed Tomography, 3D imaging technology: Fans of hard X-rays are sent through the the object and measured afterwards. Through a filtered back projection (for details see [49], ch. 3) a single slice is computed. By repeated calculation of different slices a volume can be obtained. This imaging technology is very accurate and it reveals more detailed information about soft and hard tissues than an ordinary X-Ray image. CT images show in each pixel permeability to X-rays of a voxel. Unfortunately, the dose of radiation is high.

(4) **Fiducial**
Artificial marker, that supports computer vision systems to use algorithms for faster and more accurate results.

(5) **Framegrabber**
A framegrabber is an electronic device used to discretize analog image signals to digital image. Framegrabbers commonly write via DMA (direct memory access) into the memory of a computer.

(6) **HMD**
Head mounted displays, generic term: HWD (head worn displays) that includes goggles, too: Displays that are placed in front of each eye to project images directly into the user's view. In *The Ultimate Display* [80] Ivan Sutherland presented his vision of a a perfect display. Only two years later he was the first to present a prototype of an HMD. The display which has been invented for helicopter simulators got the nickname "Damocles' sword" because of its threatening bulkiness.

There are two different kinds of HMDs (see figure 8 on page 43). Optical see-through HMDs project the images onto a half-transparent mirror, which means the real world can still be seen behind the artificial image. Video see-through displays are opaque. For these displays the view of the real world

has to be captured by a camera and combined with the artificial image. (See figure 15 and 16, on page 57 for an example of an video-see through HMD as used in RAMP).

(7) **Interlacing**
Interlacing is a common technique to reduce flickering for our eyes without increasing the update frequency. When taking or displaying the lines are not scanned subsequently but every other line. First lines with an even number are scanned and lines with an odd number afterwards, or the other way around. By this means the eye has the impression of a doubled update rate but the technical advantages of a lower update rate like longer shutter times and a lower data rate remain.

(8) **Proprioceptor**
'A sensory receptor, found chiefly in muscles, tendons, joints, and the inner ear, that detects the motion or position of the body or a limb by responding to stimuli arising within the organism.' American Heritage Dictionary of the English Language, Fourth Edition

(9) **MRI**
Magnetic Resonance Imaging, 3D imaging technology: As in CTs[3] slices are calculated from measurements taken circularly, but the medium is magnetism in contrast to radiation. The object is inserted into a strong permanent magnetic field. This field is changed by a short impulse of a strong electric magnet. With a certain probability water molecules in the material change their direction because of their own magnetic nature. Rotation of magnets induces an electric field that can be measured. Eventually MRIs reveals in each voxel the density of water, but also other magnetic material. An advantage of this imaging technology is that there is no radiation and some cancerous tissues differ from their healthy counterparts by water density. As disadvantages, MRI is more inaccurate than CT, it is very expensive, blood vessels might appear distorted because of iron in blood cells and last but not least no magnetic nor electronic material can be allowed in the subject or the vicinity. Furthermore MRI is time-consuming.

(10) **RAMP**
Real Time Augmentation for Medical Procedures, Augmented Reality Project at SCR[11].

(11) **SCR**
Siemens Corporate Research, founded in 1977, sub department of CT (Cor-

porate Technology) - the research and development department of Siemens AG, research interests in imaging and software engineering, for more information refer to www.scr.siemens.com.

(12) **Ultrasonography**
Ultrasonography is a 2D imaging technology. An ultrasound probe sends acoustic signals into the tissue and collects their reflections. The image shows a map of reflections caused by transitions between materials of different acoustic conductivity. Due to interferences of reflected signals, strong speckle noise can be seen in ultrasonography images. This kind of noise cannot be enhanced by simply recapturing the image or filtering it as can be done with Gaussian noise. This imaging technology is widespread because it is inexpensive, easy to handle and free of radiation. The images are given in real time in opposite to CT[3] MRI[9]. Because of speckle noise the image quality is very poor, so only professionals can recognize the information contained in them. Because of the speckle noise 3D ultrasonography is not in common use and automatic segmentation is very difficult for ultrasonography images.

(13) **Vestibule**
'The central cavity of the bony labyrinth of the ear or the parts of the membranous labyrinth that it contains' Merriam Webster Dictionary

References

[1] Wolfe J.W., Mathis D., Sklair C., Magee M.: *The Perspective View of Three Points* ,IEEE Transactions On Pattern Analysis and Machine Intelligence, vol. 13, no. 1, January 1991

[2] Haralick R.M., Lee C.N., Ottenberg K., Nölle M: *Review and Analysis of Solutions of the Three Point Perspective Pose Estimation Problem*, International Journal of Computer Vision, 12,3,331-356(1994), Kluwer Academic Publishers, Netherlands

[3] Faugeras, O.: *The Three Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press Cambridge, MA 1993

[4] Heikkilä, J.; Silvén, O.: *A Four Step Camera Calibration Procedure with Implicit Image Correction*, in Proceedings of the Conference on Computer Vision and Pattern Recognition [IEEE97], S.1106-1112

[5] Pollefeys M., Van Gool L.: *Self Calibration and Metric Reconstruction in spite of Varying and Unknown Intrinsic Camera Parameters*, In Proceedings of the International Conference on Computer Vision, 1998.

[6] More, J.: *The Levenberg- Marquardt Algorithm, Implementation and Theory*, in Watson, G.: *Numerical Analysis*, Nr. 630 in Lecture Notes in Mathematics, Springer Verlag 1977

[7] Tsai, R.Y.: *A versatile Camera Calibration Technique for High Accuracy 3D machine Vision Metrology Using Off-the-shelf TV Cameras and Lenses*, IEEE Journal of Robotics and Automation, April 1987, S.1999-2006

[8] Fischler M.A., Bolles R.C.: *Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image*, Analysis and Automated Cartography, Commun. ACM 24, no. 6, 1981

[9] Vogt, S.: *Anwendung von Stereoverfahren zur Verdeckungsberechnung im Bereich Erweiterte Realität*, Diplomarbeit, Mai 2001

[10] Wloka M.M. : *Resolving Occlusion in Augmented Reality*, In Symposium on Interactive 3D Graphics Proceedings, pages 5–12, New York, April 1995. ACM Press

[11] Proceedings of the First International Workshop on Augmented Reality (IWAR 98), San Francisco 1998

[12] Proceedings of the Second IEEE and ACM International Workshop on Augmented Reality (IWAR 99), Los Alamitos, Calif., 1999

[13] Proceedings of the IEEE, ACM and Eurographics International Symposium on Augmented Reality ISAR, Munich, Germany, October 2000

[14] Proceedings of International Symposium on Mixed and Augmented Reality ISMAR 2001, New York

[15] International Symposium on Mixed and Augmented Reality ISMAR 2002, Heidelberg, Germany

[16] Azuma R.T.: *A Survey of Augmented Reality*, Presence: Teleoperators and Virtual Environments 6 (1997), S.355-385

[17] Azuma R.T.: *Recent Advances in Augmented Reality*, IEEE Computer Graphics and Applications,November/December 2001

[18] Brügge B., Dutoit A.H.: *Object-Oriented Software Engineering. Conquering Complex and Changing Systems*, Prentice Hall, Upper Saddle River, NJ, 2000

[19] DeMenthon D.F., Davis L.S.: *Model based object pose in 25 lines of code*, International Journal of Computer Vision, 15:123–141, 1995.

[20] Horn B.K.P.: *Relative Orientation Revisited*, In Journal of the Optical Society of America A., Vol. 8, Number 10, pages 1630-1638, October 1991.

[21] Horn, B.K.P.: *Tsai's camera calibration method revisited*, tech. rep., Department of Electrical Engineering and Computer Science Massachusetts Institute of Technology, 2000

[22] Horn, B.K.: *Robot Vision*, The MIT Press, Cambridge, MA, 1986

[23] Haralick R.M., Shapiro L.G.: *Computer Robot Vision, vol.II*, Addison Wesley, Reading, MA, 1993

[24] Kato H., Billinghurst M.: *Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System*, in IWAR 1999 [12] San Francisco

[25] Press, W.H. et al.: *Numerical Recipes in C: The art of scientific computing*, Cambridge University Press 2nd ed. 1992

[26] Simon, G., Fitzgibbon, A.W., Zisserman, A.: *Markerless Tracking using Planar Structures in the Scene*, ISAR 2000 [13]

[27] Tyceryan, M., Navab, N.: *Single Point Active Alignment Method (SPAAM) for Optical See- Through HMD Calibration*, ISAR 2000 [13]

[28] Schmidt J. Vogt S.: *Dense Disparity Maps in Real-Time with an Application to Augmented Reality*, WACF 2002

[29] Gordon G., Billinghurst M., Bell M., Woodfill J., Kowalik B., Erendi A., Tilander J.: *The Use of Dense Stereo Range Data in Augmented Reality*, ISMAR 2002 [15]

[30] Hoff W.A., Nguyen K., Lyon T.: *Computer vision-based registration techniques for augmented reality*, Proceedings of Intelligent Robots and Computer Vision XV, SPIE Vol. 2904, Nov 18-22, 1996, Boston, MA, pp. 538-548.

[31] Hajnal J.V., Hawkes D.J., Hill D.: *Medical Image Registration (Biomedical Engineering Series)*, CRC Press 2001

[32] Stroustrup, B.: *The C++ programming language, The C++ Programming Language (Third Edition and Special Edition)*, Addison-Wesley 1997

[33] Meyers S.: *Effective C++: 50 Specific Ways to Improve Your Programs and Design (2nd Edition)*, Addison Wesley 1997

[34] Meyers S.: *More Effective programming with C++*, Addison-Wesley Pub Co; 1st edition (December 1995)

[35] Anton H., Rorres C.: *Elementary Linear Algebra*, Applications Version eighth Edition, Wiley & Son Inc.

[36] Foley J.D., van Damme A., Feiner S.K., Hughes J.F.: *Computer Graphics Principles and Practice*

[37] Drascic D., Milgram P.: *Perceptual Issues in Augmented Reality*, SPIE Volume 2653: Stereoscopic Displays and Virtual Reality Systems III,San Jose, California, USA, January - February 1996,pp 123-134

[38] Boyd D.: *Depth Cues in Virtual Reality and Real World*, Honors thesis for BA at Brown University

[39] Kolasinski E.M.: *Simulator Sickness in Virtual Environments*, Technical Report 1027, Army Project Number 2O262785A791 - Education and Training Technology, May 1995

[40] Forbes K.: *An Inexpensive, Automatic and Accurate Camera Calibration Method*, In Proceedings of the Thirteenth Annual Symposium of the Pattern Recognition Association of South Africa (PRASA 2002), November 2002.

[41] Xu J., Fang Z., Malcolm A., Wang H.: *A Robust Close-Range Photogrammetric System for Industrial Metrology*, Seventh International Conference on Control, Automation, Robotic and Vision (ICARCV 2002), Singapore

[42] Woo M., Neider J., Davis T., Shreiner D., *OpenGL 1.2 Programming Guide*, Addison Wesley, Reading, MA, 3rd ed., 1999 (Online Version of 'OpenGL Programming Guide' at fly.cc.fer.hr/∼unreal/theredbook/

[43] Naimark L., Foxlin E.: *Circular Data Matrix Fiducial System and Robust Image Processing for a Wearable Vision-Inertial Self-Tracker*, ISMAR 2002 [15]

[44] Milgram P., Kishino F.: *A Taxonomy of Mixed Reality Visual Displays*, IEICE Trans. Information Systems, vol.E77-D, no. 12, 1994, pp. 1321-1329.

[45] Sundarewaran V., Behringer R.: *Visual Servoing-based Augmented Reality*, IWAR98 [11]

[46] Thomas B., Piekarski W.: *ARQuake: The Outdoor Augmented Reality Gaming System*, Communications of the ACM, 2002 Vol 45. No 1, pp 36-38

[47] Lyons K., Gandy M., Starner T.: *Guided By Voices: An Audio Augmented Reality System*. Proceedings of the Sixth International Conference on Auditory Display ICAD 2000, April 2000.

[48] Kalman R.E.: *A New Approach to Linear Filtering and Prediction Problems*, Transactions of the ASME-Journal of Basic Engineering, 82 (Series D): 35-45, 1960.

[49] Kak A., Slaney M.: *Principles of Computerized Tomographic Imaging*, IEEE Press 1987

[50] Zimbardo P.G.: *Psychologie*, 6. Auflage, Springer Verlag Berlin Heidelberg, 1995, Original title: *Psychologie and Life*, Zimbardo Inc. Scott Foresmann and Company, Glenview, Illinois

[51] Faller A., Schünke M.: *Der Körper des Menschen*, 12.Auflage, Georg Thieme Verlag 1995

[52] Caudell T., Mizell D: *Augmented Reality: An application of heads-up display technology to manual manufacturing processes*. In Proceedings of the Hawaii International Conference on System Sciences, pages 659-669, 1992.

[53] State A., Hirota G., Chen D.T., Garrett W.F., Livingston M.A.: *Superior Augmented Reality Registration by Integrating Landmark Tracking and Magnetic Tracking*, Proceedings of SIGGRAPH 96 (New Orleans, LA, 4-9 August 1996), 429-438. www.cs.unc.edu/∼us/hybrid.html

[54] You S., Neumann U.: *Fusion of Vision and Gyro Tracking for Robust Augmented Reality Registration*, Proceedings of IEEE VR 2001,

[55] Wagner M.: *Design, Prototypical Implementation and Testing of a Real-Time Optical Feature Tracker*, Diplomarbeit TU München 2001 Related: DWARF Hompage www.augmentedreality.de

[56] Horaud R., Conio B., Leboulleux O.: *An analytic Solution for the Perspective 4-Point Problem*, Computer Vision, Graphics and Image Processing 47, 33-44 (1989), Academic Press

[57] Auer T.: *Dissertation*, TU Graz, March 2000.

[58] Auer T., Pinz, A.: *Building a Hybrid Tracking System: Integration of Optical and Magnetic Tracking*, in IWAR 1999 [12]

[59] Sauer F., F. Wenzel, S. Vogt, Y. Tao, Y. Genc, and A. Bani-Hashemi: *Augmented Workspace: Designing an AR Testbed*, ISAR 2000, pages 47-53.

[60] Sauer F., Wenzel F., Vogt S., Tao Y., Genc Y., Bani-Hashemi A.: *Augmented Reality Visualization of Ultrasound Images: System Description, Calibration, and Features*, ISMAR 2001 [14], pages 30-39.

[61] Sauer F., Khamene A., Vogt S.: *An Augmented Reality Navigation System with a Single-Camera Tracker: System Design and Needle Biopsy Phantom Trial*, in Proceedings of Medical Image Computing and Computer-Assisted Intervention, MICCAI 2002, pp 116-124.

[62] Sauer F., Khamene A., Bascle B., Vogt S., Rubino G.J.: *Augmented Reality Visualization in iMRI Operating Room: System Description and Pre-Clinical Testing*, Medical Imaging 2002: Visualization, Image-Guided Procedures, and Display, Seong K. Mung, Editor, Proceedings of SPIE Vol. 4681 (2002).

[63] Vogt S., Khamene A., Sauer F., Niemann H: *Single Camera Tracking of Marker Clusters: Multi-parameter Cluster Optimization and Experimental Verification*, ISMAR 2002 [15], pp 127-136.

[64] Lin Chai, William A. Hoff, Tyrone Vincent: *3-D Motion and Structure Estimation Using Inertial Sensors and Computer Vision for Augmented Reality*, Presence 2000: Teleoperators and Virtual Environments

[65] Breen D., Rose E., Klinker G., Koller D.: *Real-time Vision-Based Camera Tracking for Augmented Reality Applications*, in the Proceedings of the Symposium on Virtual Reality Software and Technology (VRST-97), Lausanne, Switzerland, September 15-17, 1997

[66] Hoff W.A.: *Fusion of Data from Head-Mounted and Fixed Sensors*, IWAR 1999 [12]

[67] Dorfmüller K.: *Robust tracking for Augmented Reality Using Retroreflective Markers*, Computers & Graphics 23 (1999) 795-800

[68] Webster A., Feiner S., MacIntyre B., Massie W., Krueger T.: *Augmented Reality in Architectural Construction, Inspection, and Renovation*, in Proceedings of Third ASCE Congress for Computing in Civil Engineering, June 17-18, 1996, Anaheim, CA

[69] G. Klinker: *Confluence of Computer Vision and Interactive Graphics for Augmented Reality*, Presence: Teleoperators and Virtual Environments, 6(4), 1997, pp. 433–451.

[70] Stricker D., Kettenbach T.: *Real-time and Makerless Vision-Based Tracking for Outdoor Augmented Reality Applications*, In International Symposium on Augmented Reality (ISAR 2001), New York, NY, October 29-30, 2001.

[71] Faugeras O., Luong Q.-T., Maybank S.J: *Camera Self calibration: Theory and Experiments*, In Proceedings of the Second European Conference on Computer Vision, Santa Margharita Ligure, Italy, 1992.

[72] Nojima T., Sekiguchi D., Inami M., Tachi S.: *The SmartTool: A system for augmented reality of haptics*, IEEE Virtual Reality Conference 2002 March 24 - 28, 2002 Orlando, Florida

[73] Mackay W.E., Fayard A.-L., Frobert L., Médini L.: *Reinventing the Familiar: Exploring an Augmented Reality Design Space for Air Traffic Control* In Proceedings of CHI98, Los Angeles, CA:ACM.

[74] Ohbuchi R., Chen D., Fuchs H.: *Visualization in Incremental Volume Reconstruction and Rendering for 3D Ultrasound Imaging*, Visualization in Biomedical Computing 1992. Chapel Hill, NC: SPIE, 1992. 1808: 312-323.

[75] Feiner S., MacIntyre B., Höllerer T., Webster A.: *A Touring Machine: Prototyping 3D Mobile Augmented Reality Systems for Exploring the Urban Environment* Proceedings of the First International Symposium on Wearable Computers (IWSC 97), IEEE CS Press, Los Alamitos, Calif., 1997, pp. 74-81.

[76] Quan L., Z. Lan: *Linear N-Point Camera Pose Determination*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No. 8, August 99.

[77] Ribo M., Pinz A., Fuhrmann A.L.: *A new Optical Tracking System for Virtual and Augmented Reality Applications*, In Proceeding of IEEE Instrumentation and Measurement Technology 2001

[78] Gelenbe E.: *Cooperating Robots in Augmented Reality Third Monthly Report*, Submitted to STRICOM September, 2002

[79] Boyd D.: *Depth Cues in Virtual Reality and Real World*, Honors thesis for BA at Brown University, 2000

[80] Sutherland I.: *The Ultimate Display*, Proceedings of IFIP Congress, pp. 506-508, 1965.

[81] Drascic D., Grodski J.J.: *Defence Teleoperation and Stereoscopic Video*, Proceedings of SPIE Vol. 1915, Stereoscopic Displays and Applications IV, pages 58-69, San Jose, California, Feb 1993.

[82] Weisstein, E.W.: *Eric Weisstein's World of Mathematics*, Wolfram Research, 2003, mathworld.wolfram.com Rodrigues' Rotation Formula: mathworld.wolfram.com/RodriguesRotationFormula.html

[83] Kaiser P.K.: *Joy of perception: A Web book.*, York University http://www.yorku.ca/eye/thejoy.htm

[84] Nave C.R.: *Rods and Cones*, Physics textbook of Georgia State University http://hyperphysics.phy-astr.gsu.edu/hbase/vision/rodcone.html

[85] A.R.T. GmbH. Product: Infrared tracking http://www.ar-tracking.de/

[86] BrainLAB. Product: VectorVision - Passive marker tracking http://www.brainlab.com

[87] Nexgen Ergonomics. Product: Optotrak - Active marker tracking http://www.nexgenergo.com

[88] Stan Birch's Hompage. *Introduction to Projective Geometry*, http://robotics.stanford.edu/ birch/projective

[89] Canadian Mind Products Website. *How to write unmaintainable code*, http://mindprod.com/unmain.html

[90] U.S. Navigation Center Hompage. *Information about GPS*, http://www.navcen.uscg.gov/

[91] ESA Galileo Internetsite. *Information about Galileo* http://www.esa.int/navigation/galileo