Integration of a Time-of-Flight Camera into a Mixed Reality System for Handling Dynamic Scenes, Moving Viewpoints and Occlusions in Real-Time

Bogumil Bartczak, Ingo Schiller, Christian Beder and Reinhard Koch
Computer Science Department
Christian-Albrechts-University Kiel
Hermann-Rodewald-Str. 3
24118 Kiel, Germany
{bartczak,ischiller,beder,rk}@mip.informatik.uni-kiel.de

Abstract

We present a novel approach to mixed reality applications. The key characteristics of the presented system are the use of a automatically generated static environment model and a time-of-flight camera device. The combination of both allows the correct handling of mutual occlusion between real and virtual content on the fly, which is not possible with the currently applied approaches. Typically expensive studio setups with complex camera tracking installations and multi-camera approaches in combination with chroma-keying facilities are used. Our system is rather inexpensive, compact, mobile, flexible and provides convenient calibration procedures. The use of a background model not only eliminates the need for chroma-keying in mixed reality production, it moreover supports planing and alignment of virtual content. Based on depth information the system is generating appropriate depth maps in real-time, making the approach suitable for 3D-TV productions. The presented paper discusses all key elements of mixed-reality applications based on our approach. This includes camera pose tracking, correct real-time handling between interacting virtual and real content and the fast environment model building.

1. Introduction

The stunning impressions nowadays film and TV productions of fictive events convey to the spectator, is owed to the combination of real imagery and computer graphics generated virtual content. For a proper augmentation of reality through virtual content some challenges have to be tackled. On the one hand a temporal consistent alignment between rendered virtual content and the real image footage has to be established. On the other hand mutual occlusions of the virtual and the real world objects have to be handled correctly. Finally lighting conditions and optical effects like shadows and reflections have to be matched between both “worlds”. In [18] a good overview of the current techniques for mixed reality application in TV and film productions is given. In there Thomas also discusses the usefulness of real-time capable approaches, which do not deliver the final result but close on-set previews for actors and the production crew. In this paper we present such an approach to real-time mixed reality applications based on a background environment model and the use of a time-of-flight (ToF) camera.

Our approach is based on a background environment model, which in our understanding captures the static geometric and photometric properties of the scenery, in which the data acquisition is performed. Such a model can aid in the production of mixed reality data in various aspects. Typically “virtual studios” equipped with chroma or difference keying facilities and expensive camera tracking installations, like special marker setups [19] or camera mounts with pan-tilt sensors, are used. The tracking of the camera parameters is needed for the aligned rendering of the virtual content to create an illusion of the fictive content also being part of the scenery. This implicates that the coordinate frame used for tracking (i.e. system of markers) is properly aligned with the virtual objects’ coordinate frames and the interacting real content. A small inconsistency in this calibration can already destroy the targeted illusion.

In [5, 8] systems are proposed, which track a camera based on such a background environment model. The digital representation of the model not only eases the task of aligning the virtual data to the real data, it then also solves the consistency problem if the used model is sufficiently precise. Indeed, the model generation is the critical ingredient in this kind of systems. For the purpose of tracking purely 2D vision based scene reconstruction can be applied.
for the background model creation. In [1] such an approach using a fisheye camera for the consistent reconstruction of large scenes is discussed. The system is flexible, compact and very economic. In order to generate reliable high resolution data many images with different baselines have to be processed using structure-from-motion, bundle adjustment and dense depth estimation. The runtime of this approach is therefore high. Moreover the necessary correspondence analysis cannot be reliably performed in texture-less regions, which leaves gaps in valuable occlusion data of particular scene parts. While structured-light approaches are not feasible for large scale reconstruction, the application of a laser-scanner presents an alternative for the static scene reconstruction. This device reliably reconstructs almost any geometric configuration regardless of texturing. The low processing time makes laser scanners a good choice, even in scenarios where the background scene is changed during production. Nevertheless these devices are expensive and do not capture the photometric information needed for a visual tracking approach. For this purpose a calibrated combination of a laser-scanner with a 2D color image camera can provide a solution (cf. [3]). Moreover a laser-scanner is not suitable for the capturing of dynamic data, therefore its usage in a mixed reality application, aside from the task of static background modeling, is limited. Recently, active depth measuring devices based on the time-of-flight (ToF) of an intensity modulated infrared light front have reached maturity (cf. [11]). The effective operation range of these ToF-Cameras is smaller than the one of laser scanners, but on the contrary these devices are capable of capturing dynamic data over an extended field-of-view. We therefore propose the application of such a ToF-camera in the presented context. In this work we show how this device can be used for fast large scale background model construction. Furthermore we exploit its depth capturing capabilities during the (online) real data capturing. Here the depth information is used in combination with the previously constructed background model to key foreground objects, virtual objects and the background on the fly. This way the need for chroma keying facilities is made obsolete. Moreover a convenient and correct live handling of mutual occlusions of virtual and real content is provided, which is not possible by means of simple chroma keying techniques.

The remainder of this paper is structured as follows. The next section presents the proposed approach as a whole. Afterwards central issues are discussed in more detail. This discussion starts with the description of the necessary calibration procedure of the used camera rig shown in figure 1. Section 2.3 then elucidates the background model creation with this camera rig. Section 3.3 to 3.5 deal with the important aspects during live processing. These are the real-time transport of captured depth data, the mixing and the segmentation of image content exploiting the processing power of modern graphics cards. Furthermore the mandatory camera pose estimation is discussed. After presenting the achieved results in section 4 we conclude this work in section 5.

2 System Overview

We will start with a brief overview of our real-time mixed reality system. An outline is shown in figure 2. The systems purpose is the generation of a mixed image and a segmentation for a target camera view, shown on the right hand side of figure 2. As can be seen in the shown target input image, no chroma keying facility is used to segment the actor. This is possible due to the use of a background model (see fig. 4) in combination with a ToF-Camera (see fig. 1). The ToF-camera delivers an instantaneous depth image, shown in the left-middle of fig. 2, of the scene but from a different viewpoint and with different intrinsic parameters than the target view camera. Therefore a warping of the depth measurements into the target view has to be performed. A convenient and very fast depth transfer is possible by means of a graphics accelerator card, which is more closely explained in section 3.3. The graphics accelerator card is also programmed to do the pixel based depth comparison for segmentation and proper mixing between real and virtual content (see section 3.4). The consequent use of graphics accelerator hardware allows to perform all computations of the online phase in real-time.

The view dependent data needed for this tasks is rendered from the virtual objects description and the back-
Figure 2. System components and interactions. The grey boxes are discussed in detail in the text.

ground model. A correct rendering is only possible if the target view camera pose is known in the model’s coordinate frame. We therefore added a wide field-of-view fisheye camera to the system, which operates as a reliable and precise pose sensor. A background model based tracking approach, elucidated in section 3.5, is then used to track the fisheye camera pose in the model’s coordinate frame. In order to derive the correct rendering pose the relative orientation and position of the pose sensor camera and the target camera is determined in an offline rig calibration procedure explained in section 3.1. This is also needed during the offline background model creation, where the ToF-Camera is systematically moved by a pan-tilt-unit (PTU) to capture the static environment. The retrieval of the environment’s geometric and photometric properties is topic of section 3.2. Once the background model has been created, the planning and alignment of virtual content can be performed by means of 3D modeling tools. The subsequent section will discuss the central components of the system (indicated as grey boxes in fig. 2) in more detail.

3. System Components

3.1. Calibration

The internal and relative calibration of the used cameras is a crucial aspect in this application. The major challenge here arises from the fact, that the system comprises of three different camera types. Those camera types are a classical perspective camera, a camera equipped with a fisheye lens and a ToF depth camera.

Well established techniques are available for calibrating classical perspective cameras including radial lens distortion. Those are usually based on multiple images of a calibration pattern. A simple and flexible method based on images of a planar checkerboard pattern has been presented for instance in [20] or [4]. For the calibration of fisheye cameras a very similar approach based on the same planar checkerboard as calibration pattern is available. This has been presented in [12] and [13]. Finally, the calibration of the ToF camera is required. Such a calibration based on a planar calibration pattern has been presented for instance in [9] or [2].

However, in order to obtain an optimal calibration for the entire system including the relative transformations between the different cameras, an overall adjustment is required. To achieve this goal, we use the analysis-by-synthesis approach presented in [14]. All camera parameters including relative transformations, internal calibration for the fisheye lens as well as for the perspective cameras, and systematic depth measurement errors of the ToF camera are used to render the known calibration pattern into each image (including the depth images of the ToF camera). Comparing those rendered images with the actual images allows to compute a residual for each pixel. Furthermore, it is possible to compute partial derivatives of the residuals with respect to each of the parameters numerically by re-rendering the images with slightly distorted parameters. This enables the minimization of the residuals using standard adjustment techniques, such as the Levenberg-Marquardt algorithm, yielding a best linear unbiased estimate of all the parameters. Note, that the huge amount of rendering operations can be efficiently performed using graphics acceleration hardware, so that reasonable running times are achieved.

3.2. Generating the Model

In this section we explain how the background model, which shall aid in the production planning, the visual camera tracking and the depth-keying, is generated utilizing the
depth capturing capabilities of the ToF-camera. Since the field of view of the ToF-camera is too small to capture a sufficiently large portion of the environment, the ToF-camera and the target camera were mounted onto a pan-tilt-unit (PTU), as can be seen in figure 1. This way the scene can be scanned with the ToF-camera collecting the necessary data for geometry reconstruction. The color camera simultaneously captures color images, which deliver the photometric information needed in the visual camera tracking. The PTU is programmed to do a panoramic sweep covering a field of 270° in horizontal and 180° in vertical direction. Rather than directly generating a 3D representation of the scanned environment, two cylindric panorama images are generated, one depth panorama and one color panorama, like shown in figure 3. The creation of this panoramas relies on the previous calibration of intrinsic parameters and the relative orientation and position of the ToF-Camera, the target camera and the tool center point of the PTU. Thereafter the correct panorama content can be calculated by the known PTU orientations and the captured depth data in a forward mapping process. The neighborhood relations in the panoramas allow to improve the data by morphological operations and median or bilateral filtering. Moreover this data structures are well suited for a straightforward handling of overlaps in the captured images during the scanning process. The generation of such a panorama on the basis of 22 captured images needs approximately 60 seconds.

In order to generate a model representation suitable for rendering, the depth- and intensity panorama is converted into a textured triangle-mesh following the proposition described in [1, p. 93]. The resulting textured triangle-mesh is shown in figure 4. Please observe the fill rate of 100% in the scanned areas, even in the lowly textured parts, which would not have been obtainable using solely image based approaches. For a reduction of the high amount of data the triangles in the surface mesh can be processed by means of the propositions presented in [15].

In [6] a comparable approaches to model building using a ToF-camera is described. In contrast to our work registration has to be done between subsequent measurements using texture and depth information combined with inertial sensor measurements. This registration is not required in our approach as the PTU delivers high precision rotation data. The generated model could however be extended by moving the camera rig to a different location. Thereafter a combination of our approach and the method in [6] could be applied to merge multiple models.

3.3. Warping the Depth Map on the GPU

The color and depth data used for the final augmentation is simultaneously captured by different devices, which do not have the same center of projection. In order to correctly perform the mixing between real and virtual data, the depth information captured by the ToF-camera has to be transferred to the target view. For this it is assumed that each pixel value observed in the image captured by the ToF-camera is closely representing the depth belonging to a single point of the 3D scene projecting itself to the pixel’s center. This 3D point is easily reconstructed using the previously acquired metric calibration of the ToF-camera. Simple scaling of the unit length optical ray vector by this depth value delivers the 3D point in the local coordinate frame of the ToF-camera. Connecting such 3D points belonging to neighboring pixels by a triangle, yields a closed surface mesh of the observed scene. This 3D surface representation can be processed with the image synthesis facilities provided by OpenGL to acquire a z-Buffer image of the ToF depth capture from the target view.

By using OpenGL the depth propagation to the target view is (more or less) automatically delegated to the graphics hardware dedicated to this task. This way a very high
performance of the depth propagation is achieved. Bottleneck in this processing chain is the exchange of data between the memory accessible by the GPU and the CPU respectively. Please observe that the intrinsic parameters of the ToF-camera are fixed. Therefore the topological information needed for the surface mesh creation in form of the optical ray vectors and the pixels belonging to the same triangle are known before the actual depth capturing. This information can be pre-loaded to the graphics card’s memory in an initial step. Afterwards the state of the art vertex shader stage is used to perform the aforementioned scaling of the optical rays to generate the correct surface geometry. By means of this procedure the performance is not only improved by limiting the needed data transfer to the captured depth image per frame, but also by the exploitation of the parallel processing power of the graphics card for the 3D point generation.

In general the camera model needed for the appropriate target view contains geometric lens distortions, which can be represented by a two dimensional displacement map [10, p.297]. In this case the depth transfer is performed in two passes. In the first pass ideal perspective parameters are used to render an undistorted z-Buffer map, by means of the vertex shader based rendering described above. The second pass is creating the distorted result by using a fragment shader stage for indirect look-up into the previously generated z-buffer map via the displacement map containing the distortion information. Again the intrinsics of the target view do not change and thus the necessary distortion information can be upload to the graphics card before live processing. State of the art render-to-texture facilities permit all calculation to solely use the GPU’s memory domain. A transfer of the intermediate results to main memory is not needed, because the foreground segmentation and augmentation is completely performed on the graphics card, like described in the following section.

3.4. Final Depth Layer Generation and Image Mixing

As stated before the goal of the system described in this work is to incorporate virtual 3D data content into a live video stream. But in contrast to simple augmentation, a correct real-time handling of occlusions between dynamic virtual and dynamic real content shall be provided. Additionally a separation of the dynamic real foreground objects from the static background scene shall be performed.

In the previous section the depth transfer from the ToF-camera view to the targeted augmentation view was discussed. The result is a z-Buffer image of the surface observed by the ToF-camera. Using the previously described two-pass distortion rendering approach, two more z-Buffer images are generated for the target view. One z-Buffer image is originating from the background model (see section 3.2) and the second z-Buffer image is based on the virtual objects that have to be inserted. A final rendering pass is used to process the augmentation and segmentation based on this depth information on a pixel basis. A comparison between the z-Buffer image calculated from the background model and the z-buffer from the ToF-depth warping enables to decide, which pixels in the target view belongs to the background and which image part is related with (dynamic) foreground. A manually set threshold has to separate the z-Buffer values in order to assign a pixel to the foreground. This threshold compensates smaller calibration errors and noise in the ToF-depth image. The z-Buffer image of the virtual objects can be compared to the foreground depth as well as the background depth so that background and foreground objects correctly are hidden or occlude the virtual objects. Within a fragment program different depth comparisons can be performed simultaneously. Modern graphics cards are moreover capable of delivering multiple results into different images within a single rendering pass. Fig. 8 shows the results achieved with a shader setup that on the one hand separates the dynamic real foreground pixels from the background and mixes them with the virtual content. On the other hand in fig. 9 the background pixels are filled with the current target image, creating the impression that the virtual content is standing in the room and the person is moving around it. Of course the shader performing this separation and final image composition is additionally passed the rendered intensity images of the virtual objects and the currently captured target view image.

3.5. Acquiring the Camera Pose

The previously discussed calibration of the used camera rig (see section 3.1) delivers the intrinsic calibration of all cameras in the rig and their relative extrinsic relation. While this fixed calibration is sufficient for the purpose of depth warping, the rendering of the background model and virtual objects requires the target camera’s current pose in the respective coordinate frame. Aligning the virtual objects’ coordinate frames to the background-model’s coordinate frame in an offline processing step, leaves the system with the task to determine the camera pose in the background model frame. This has to be done with a sufficiently high framerate and while dynamic objects move through the scenery. It was found that the analysis-by-synthesis camera tracking approach presented in [8] has the necessary properties to tackle this task. The proposed approach uses a freeform background-model as an absolute reference for pose estimation within the model’s coordinate frame. By means of the combination between the absolute reference and a fisheye camera the pose estimation is very reliable [17] and free of error accumulation (drift). Rather than globally opti-
mizing the intensity difference of the currently captured image data and the reference data for pose estimation, the system maintains a set of 3D points generated from the background model at interest points ([16]). These 3D points are independently tracked from a synthesized view of the reference model to the currently captured image. Based on the established 2D-3D-correspondences the current pose can be estimated. Before a feature contributes to the estimation its validity is checked by a robust photometric measure in order to detect features, which are occluded by a dynamic object. The fisheye camera’s extended FoV always provides sufficient visible features for reliable tracking, even if large parts of the used background model are occluded in the targeted camera view. Utilizing the previously determined fixed relation between target camera and fisheye camera within the used rig, the determined pose can be mapped to the representation needed for rendering.

4. Results

Evaluating the system with synthetic data produces reasonable results. Occlusion detection and mixing of perfect models and depth measurements is not a challenge. The challenge here is the handling of noise during the segmentation and mixing and the alignment of the generated model to the current image frame. Therefore we decided to present the evaluation of the whole system qualitatively on real data. The data used in the shown case consists of 420 images taken simultaneously with each of the three cameras, the ToF-, the perspective CCD- and the fisheye-camera. The input images of the perspective camera, also called the target camera in the previous sections are shown in figure 5.

As described in section 3.3 the depth images, acquired from the ToF-camera are warped to the projection of the target camera. Figure 6 shows the warped depth maps as z-buffer values. They are scaled for visibility. Observe, that the depth values fit the corresponding images of the target camera seen in figure 5 including the lens distortion effects.

These warped depth maps are mixed on the GPU as described in section 3.3 with the background model and the virtual objects which are to be rendered in the final augmented scene. The result of the mixing can be seen in figure 7. Note that at the border of the images (compare with the images in figure 6) the background model is inserted and in the middle a virtual object, the Statue of Liberty, is rendered in the z-buffer.

One possible result of the whole effort is shown in figure 8. In this depth-keying application it is possible to segment the current image in a way that only the virtual objects and the objects which are added after the background model creation are present. Here also the correct segmentation of the floor is noteworthy. This form of precise segmentation is not possible with the depth interval approaches proposed in [7]. The background is now easily replaced by for example another virtual background or a live video stream.

Note that some errors remain in the segmented images. This is due to the scattering and noise on the ToF-images. These however can be either manually removed in a post-processing step or can be overcome by content aware filtering operations.

In figure 9 the final mixing result in the target camera is presented. Note the high quality of the occlusion detection. For example the correct segmentation between the legs in the middle right image or the fine silhouette in the top right image. The bottom two images show the mixing with a moving camera. Note the changing rotation of the virtual
5. Conclusions

This work presented a novel approach to mixed-reality applications. The central issue is the use of an environment model and the integration of a ToF depth camera. It has been shown that by means of this two combined techniques a very convenient approach to the solution for various challenges in mixed-reality applications is provided. Namely these are the reliable real-time camera pose estimation despite the presence of dynamic objects contained in the images. Moreover the on the fly correct handling of mutual occlusions of virtual and real objects by means of precise depth-keying. This way no chroma-keying facilities are needed and the limitations of segmentation approaches based on depth intervals are overcome. The proposed capturing installation consists of a compact, mobile rig using only visual devices. It was discussed how this way convenient quasi-automatic computer vision based techniques can be applied to perform necessary calibrations. It was then explained how the environment model can be built utilizing this camera rig. The ToF-camera herein plays an important role, since it allows to reconstruct the environment even in areas with insufficient texturing for vision based reconstruction. The model generation takes mere minutes to complete and thus is suitable for changing production environments. The created model can be used for planning and alignment of the virtual content. Since the model is used in the online camera pose tracking, this way an ad hoc virtual content consistency is established. It was moreover discussed how modern commodity graphics acceleration cards are integrated in the process to provide the necessary processing speed in the face of depth image warping, camera distortions and pixel based segmentation. Finally the sys-
tems successful application in an all-day (uncontrolled) environment was presented.

The system’s ability of segmenting dynamic image parts from static scene parts by depth is especially useful in mixed reality scenarios. Also other applications, like security, human interface and motion capturing scenarios can profit from this information. Observe that not only reliable segmentations are provided, but also mixed and segmented depth maps are delivered by the system in real time, which is imperative for upcoming future 3D TV applications.

Acknowledgements

This work was partially supported by the German Research Foundation (DFG), KO-2044/3-1 and the Project 3D4YOU, Grant 215075 of the Information Society Technologies area of the EU’s 7th Framework programme.

References


