User-Friendly Integration of Virtual Objects into Image Sequences with Mosaics

Hellward Broszio, Thorsten Thormählen, Patrick Mikulastik

{broszio,thormae,mikulast}@tnt.uni-hannover.de
University of Hannover, Information Technology Laboratory, Hannover, Germany

Abstract

In this paper an automatic system to integrate virtual objects into image sequences taken by a rotating and zooming camera is presented. In this case of camera motion the observed static scene can be represented as a 2D panoramic mosaic image, which is estimated from the image sequence. Virtual objects can be positioned easily in the mosaic image with standard image editing software. The modifications made on the mosaic image are applied to all images of the sequence by the system. Occlusions of virtual objects by moving foreground objects of the real sequence are taken into account. In contrast to existing computer vision approaches the proposed system is very easy to handle, because no 3D modelling and animation tool is required for positioning of virtual objects in the 3D geometry of the observed scene. If moving foreground objects from the real sequence occlude virtual objects, usually the required image masks have to be generated manually. The presented system calculates these masks automatically.

Keywords: Augmented reality, virtual objects, camera tracking, panoramic mosaic image, compositing, video processing.

1 Introduction

The integration of virtual objects into image sequences captured by a moving camera is of high interest for special effects in TV and movie productions. A virtual camera generates synthetic images of virtual objects, which are mixed into the real camera images. The goal is the creation of an augmented image sequence with the illusion that the virtual objects are part of the scene taken by the real camera. The integration is associated with four main processing steps:

1. The camera parameters of the real camera must be measured or estimated and transferred to the virtual camera.
2. The generation of the 3D geometry of the static real scene and its 3D moving objects.
3. The positioning of the virtual objects in the 3D geometry.
4. The generation of the new image sequence considering partial or full occlusions of virtual objects with moving objects.

Currently used systems can be divided into broadcast and post production applications.

In broadcast applications, e.g. PVI's "L-VIS" or Sportvision's "1st and Ten", camera parameters are measured with sensors. The sensors are mounted at the cameras and record the camera parameters associated with the sequence. The 3D geometry of the static scene is generated manually offline. In this 3D geometry the virtual objects are positioned and synthetic images are generated by a virtual camera. In the final processing step for each pixel of the original image sequence the decision is made, whether its color has to be replaced by the color from the synthetic image or not. Whereby the decision depends on a lookup table, which holds information about the color of the moving foreground objects from the real scene, that cause occlusions.

In post production applications camera parameters are mostly estimated from image information with computer vision approaches, which were published for general camera motion, e.g. [4, 9], and for the special case of a rotating and zooming camera, e.g. [4, 7]. The 3D geometry can be generated either with laser scanner, e.g. I-SiTE's "3D laser imaging system", or with computer vision techniques, e.g. [6]. The positioning of virtual objects in the 3D geometry can be achieved with 3D modelling and animation tools, e.g. Avid's "Softimage". Compositing tools, e.g. Adobe's "AfterEffects", are used to generate the final augmented image sequence, where they also consider occlusions of virtual objects occurring from moving objects of the real sequence by manually generated image masks.

The special hardware equipment required for this post production technique is expensive and complicated to handle. It restricts the integration of virtual objects to sequences, which are captured by this special hardware.

Also the lookup table, that holds color information about the moving foreground objects from the real sequence have to be readapted manually.

The positioning of virtual objects in the 3D geometry is associated with high effort and thus can only be realized efficiently by users who are familiar with 3D modelling and animation tools. Current computer vision solutions, which acquire 3D geometry of moving objects from image sequences with camera motion, are not yet able to calculate occlusions with the desired accuracy. Hence, for calculation of occlusions compositing tools are used, but these are associated with a lot of manually interventions to generate the label masks and also require an experienced user.

This paper address these problems and presents a user-friendly automatic system to integrate virtual objects into image sequences taken by a rotating and zooming camera mounted on a tripod. It requires no special hardware equipment, because only image processing techniques are used. Furthermore, the system avoids the positioning of virtual object in the 3D geometry and generates automatically the label masks to consider occlusions. This is achieved because the 3D geometry can be represented by a high-resolution mosaic image in the case of a rotating and zooming camera. By using a mosaic the positioning of virtual objects is simplified to 2D image editing. Also no compositing tool must be employed, because the occlusions are detected from differences between the real image sequence and the decomposed mosaic image. Fig. 1 illustrates the components of this system, where every component corresponds to one section of this paper. Although the components are partially known, a system with this kind of component interaction has not been proposed before.

In Section 2 a robust method to estimate camera parameters from corresponding image points is described. In Section 3 the generation of high-resolution mosaic images is presented. The positioning of virtual objects into the mosaic image is described in Section 4. Subject of Section 5 is the image generator, where the mosaic images are decomposed and merged into an augmented image sequence. Experimental results are given in Section 6 and the last section provides a brief conclusion and discussion.

2 Robust Estimation of Camera Parameters

The estimation method consists of three processing steps: detection of feature points, determination of correspondences and robust estimation of the camera parameters.

Using homogeneous representation of coordinates, a 3D point is represented as $\mathbf{X} = (X, Y, Z, 1)^\top$ and a 2D image point as $\mathbf{x} = (x, y, 1)^\top$.

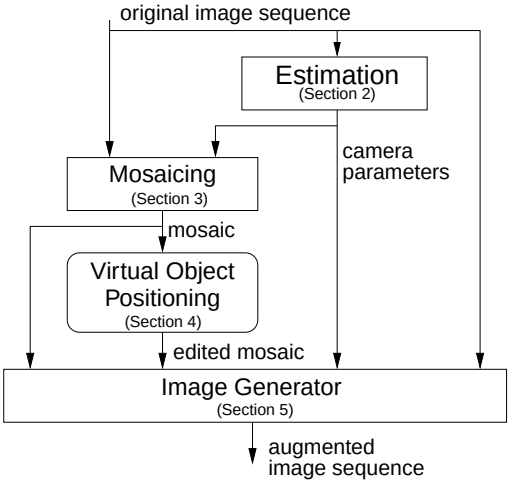

Figure 1: An overview of the proposed system, showing the connections between components of the system.

2.1 Detection of Feature Points

The feature points are detected with subpixel accuracy using the Harris feature point detector [3]. For each image of the sequence a list of feature point coordinates $L = \{\mathbf{x}_1, \ldots, \mathbf{x}_i, \ldots, \mathbf{x}_M\}$ is extracted.

2.2 Correspondence Analysis

The feature points in list L and L' of two successive images are assigned by measuring normalized cross-correlation between 15×15 pel windows surrounding the feature points. The correspondences are established for those feature points, which have the highest cross-correlation. This results in a list of correspondences $L_c = \{q_1, \ldots, q_i, \ldots, q_N\}$, where $q_i = (\mathbf{x}_i, \mathbf{x}'_i)$ is a correspondence.

2.3 Estimation of Camera Parameters

For the estimation of camera motion parameters from corresponding feature points, the real camera must be represented by a mathematical camera model.

Camera Model

The camera model (Fig. 2) describes the projection of a 3D point $\mathbf{X}$ to the image coordinate $\mathbf{x}$ through a perspective camera with

$$\mathbf{x} \sim \mathbf{K}\,[\mathbf{I}|\mathbf{0}]\,\mathbf{X} \tag{1}$$

where $\sim$ means equality up to an arbitrary non-zero scale, $\mathbf{I}$ is identity matrix and the camera calibration matrix $\mathbf{K}$ is defined by:

$$\mathbf{K} = \begin{bmatrix} f\gamma & s & c_x \\ & f & c_y \\ & & 1 \end{bmatrix} \tag{2}$$

where f represents the focal length, γ represents the aspect ratio, s represents the skew and $\mathbf{c} = (c_x, c_y, 1)^\top$ describes the principal point.

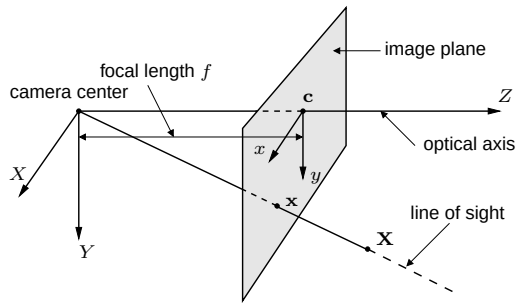

Figure 2: Camera model.

Motion Model

For the described application absence of translational motion is required. Therefore the motion model considers only camera rotation. The rotation in 3D space is uniquely given by a 3 x 3 rotation matrix $\mathbf{R} = [r_{mn}]$. The rotation matrix $\mathbf{R}$ has three degrees of freedom, which can be interpreted as three rotating angles pan φ, tilt ϑ and roll ρ, describing three successive rotations around the Y-, X- and Z-axes, respectively.

For pure rotational motion of the camera, a planar homography relates corresponding image points [5]:

$$\mathbf{x}' \sim \mathbf{K}'\mathbf{R}(\mathbf{K})^{-1}\mathbf{x} = \mathbf{H}'\mathbf{x} \tag{3}$$

To estimate camera parameters, the calibration matrix can be constrained by invariant zero-skew $s = 0$, or known pixel ratio or known invariant principle point c_x, c_y [1, 7]. All three assumptions have been made here. If there is nothing known about the calibration matrix, it is sufficient in practice to set the pixel ratio to square-pixel and the principle point to the image center. The homogeneous representation (3) can be converted to nonlinear equations, which express the relationship between the image coordinates $\mathbf{x}_i = (x_i, y_i, 1)^\top$ and $\mathbf{x}'_i = (x'_i, y'_i, 1)^\top$ of a projected 3D point.

$$\begin{aligned} x'_i &= \hat{x}'_i(x_i, y_i, \varphi, \vartheta, \rho, f, \zeta) \\ &= \zeta f \frac{r_{11}x_i + r_{12}y_i + r_{13}f}{r_{31}x_i + r_{32}y_i + r_{33}f} \\ y'_i &= \hat{y}'_i(x_i, y_i, \varphi, \vartheta, \rho, f, \zeta) \\ &= \zeta f \frac{r_{21}x_i + r_{22}y_i + r_{23}f}{r_{31}x_i + r_{32}y_i + r_{33}f} \end{aligned} \tag{4}$$

The estimation value $\hat{\mathbf{x}}'_i = (\hat{x}'_i, \hat{y}'_i, 1)^\top$ of $\mathbf{x}'_i = (x'_i, y'_i, 1)^\top$ has five unknowns: pan φ, tilt ϑ, roll ρ, focal length f and zoom $\zeta = f' \,/\, f$. In the following steps a robust and precise estimation algorithm for these five camera parameters is developed.

Outlier Detection

Due to erroneous assignment of feature points arising from moving objects or illumination changes in the scene, usually some of the correspondences are incorrect. To achieve a robust estimation of camera parameters, a random sampling algorithm [2] for outlier detection is employed to detect reliable correspondences. The random sampling algorithm uses a fast linear estimation of the homography $\mathbf{H}'$, as introduced by [8].

Only the inliers are applied to estimate the camera parameters in the following processing step.

Focal Length Estimation

The estimation of focal length f requires a sufficiently large camera rotation between two images to obtain robust and precise results. The selection of the reference image, which is used to estimate the focal length of the first image, is based on investigation of corresponding image points, which are tracked over the image sequence. The reference image is chosen, if a maximum amplitude of pan plus tilt rotation is determined, and the number of tracked inlier correspondences does not fall below a given threshold of 50%.

$$\sum_i (\Delta x'_i)^2 + (\Delta y'_i)^2 \rightarrow \min \tag{5}$$

$$\text{with} \quad \begin{aligned} \Delta x'_i &= x'_i - \hat{x}'_i(x_i, y_i, \varphi, \vartheta, \rho, f, \zeta) \\ \Delta y'_i &= y'_i - \hat{y}'_i(x_i, y_i, \varphi, \vartheta, \rho, f, \zeta) \end{aligned}$$

The five parameters pan, tilt, roll, focal length and zoom are estimated for the first and the selected reference image by using Levenberg-Marquardt optimization, which minimizes the residual error of the cost function (5) for all inliers. The estimated focal length is used to initialize the first camera of the sequence.

Rotation and Zoom Estimation

The four remaining camera parameters pan, tilt, roll and zoom are estimated similar as described before by minimizing Eq. (5), but the residuals (6) are calculated from successive images with Eq. (4).

$$\begin{aligned} \Delta x'_i &= x'_i - \hat{x}'_i(x_i, y_i, \varphi, \vartheta, \rho, \zeta) \\ \Delta y'_i &= y'_i - \hat{y}'_i(x_i, y_i, \varphi, \vartheta, \rho, \zeta) \end{aligned} \tag{6}$$

Estimation of zoom instead of focal length results in more stable results for successive images with small rotation angles.

Global Estimation

Especially for long video sequences with large rotation angles most proposed algorithms have problems with

long time stability, because estimation of camera parameters is critical concerning the accumulation of estimation errors. In contrast to other algorithms, where the parameters are estimated from image to image, the suggested approach estimates the parameters of each image of the sequence in relation to a global coordinate system. Therefore all measured feature points of inliers are registered in the global coordinate system on their first occurence. In the absence of translational motion, the global coordinate system is a sphere centered to the camera center. After each successive estimation step, all previous camera parameters and the registered feature points in the global coordinate system are used as an initial guess for a Maximum-Likelihood estimation. The Maximum-Likelihood estimation optimizes all camera parameters up to the actual image as well as the positions of the feature points in the global coordinate system.

3 Mosaicing

Using the estimated camera parameters all image points of each image are projected into a mosaic (Fig. 3). This mosaic can either be constructed as a planar or a cylindrical surface. For the projection on a planar surface the estimation of a homography would be sufficient, but the projection on a cylindrical surface requires knowledge of the camera parameters as estimated before.

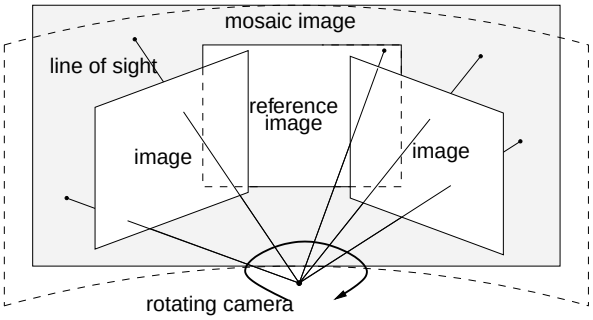

Figure 3: Assemble all image points into a mosaic.

3.1 Interpolation

Since the projection of image points results in continuous image coordinates, mosaicing requires interpolation of subpixel intensity values.

In commonly used image sequences the overlapping areas between successive images are huge. Hence, there are many intensity values M_k, projected from different images and located at subpixel position, available to interpolate the intensity values I_m of one image point on the sampling raster of the mosaic with

$$I_m = \frac{1}{\sum\limits_k a_k} \sum_k a_k M_k$$
$$a_k = \left(1 - \sqrt{2} d_k\right) \sum_j d_{kj} \tag{7}$$

where d_k is the distance of M_k to the sampling position in the mosaic and d_{kj} is the distance between sampling position M_k and M_j. Only intensity values M_k with $d_k < 1/\sqrt{2}$ pel are used to interpolate I_m (see Fig. 4).

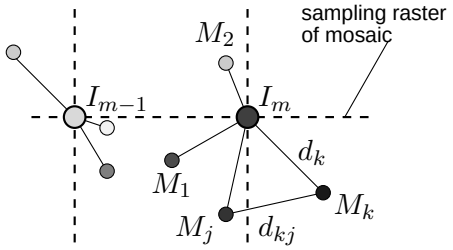

Figure 4: Interpolation of intensity values I_m on the sampling raster of the mosaic with projected intensity values M_k and distances d_k.

3.2 High-Resolution Representation

As illustrated in Fig. 5 resolution enhancement of the resulting mosaic reduces the quantization error caused by interpolation, because the mean distance d_k to the sampling raster is smaller and thus less intensity values M_k are used for interpolation. The mean quantization error is also decreased by adjusting the sampling raster of the mosaic to one reference image chosen from the sequence. With a conform sampling raster image points included from this image require no interpolation and thus produce no quantization error.

3.3 Moving Object Elimination

By analysing the relative frequency f_r of the projected intensity values M_k, it is possible to eliminate moving objects from the mosaic. Therefore, for each pixel the mean deviation σ_m of the intensity values M_k from their associated intensity values I_m is calculated. Then the arithmetic mean $\bar{\sigma}$ over all σ_m is determined. Fig. 6 shows how a $2\,\bar{\sigma}$ window is positioned, where it encapsulates the maximal number of intensity values. These encapsulated values are assumed to belong to the static background, the others to moving objects. To remove these objects, I_m is calculated only with the intensity values M_k from the static background in the interpolation step 3.1.

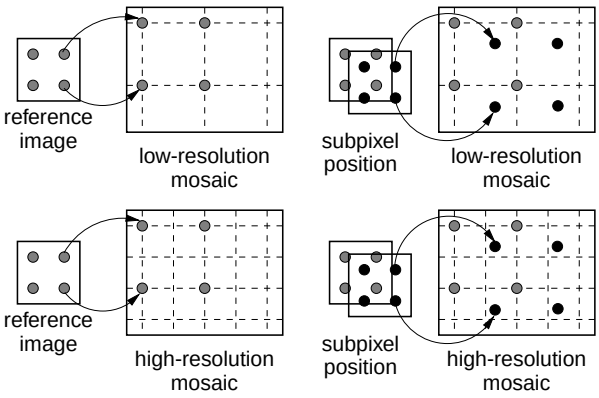

Figure 5: Reduced quantization error by resolution enhancement.

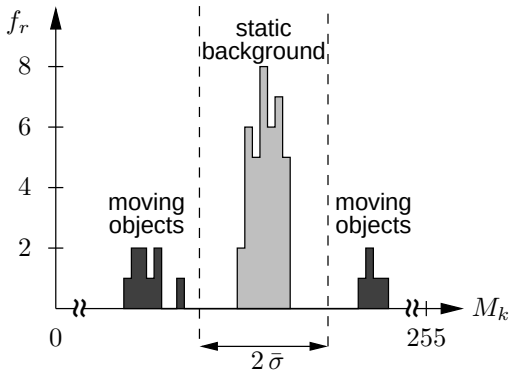

Figure 6: Analysing relative frequency f_r over M_k for each pixel I_m to separate moving object and static background.

4 Virtual Objects Positioning

The positioning of virtual objects into the original image sequence as new background objects can now be achieved by modifying the mosaic image with standard image editing software, e.g. Adobe's "Photoshop".

5 Image Generator

As seen in the system overview in Fig. 1 the final processing step is the image generator, which inputs are the original image sequence with the estimated camera parameters for each image, the mosaic image and the edited mosaic including virtual objects. The components, which are required to produce the composed image sequence, are illustrated in Fig. 7.

5.1 Mosaic Decomposition

Using the estimated camera parameters the mosaic and the edited mosaic are decomposed into two sequence of images by back-projection. The comparison of these sequences results in a edited region label mask for each image.

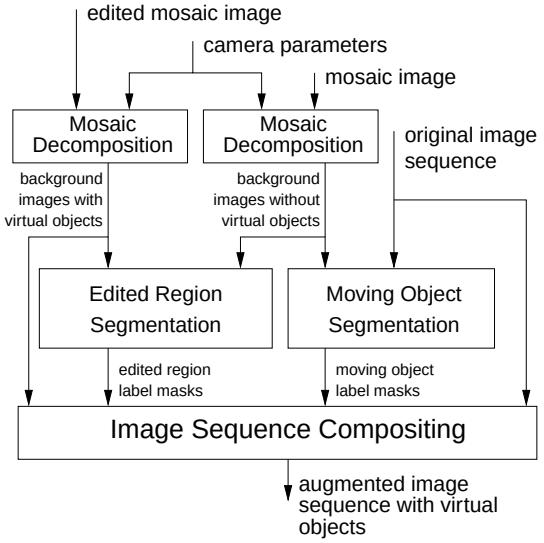

Figure 7: Generation of a augmented image sequence.

5.2 Moving Object Segmentation

The goal of placing the virtual objects onto the static background requires the recovery of moving foreground objects from the original sequence. Therefore a segmentation algorithm is employed, which calculates the difference between the original and the background images without virtual objects obtained from the unmodified mosaic. After thresholding the image difference depending on $\bar{\sigma}$, a morphological filter is applied to shape objects in the image and eliminate too small features. These regions are stored in moving object label masks.

5.3 Image Sequence Compositing

The final image sequence is composed from original images and rendered background images from edited mosaic depending on the two label masks. To avoid aliasing, these label masks are low pass filtered. An image region in the original images is replaced, if this region is marked as edited in the edited region label masks and is not determined as moving foreground object.

6 Results

Efficiency of the algorithm is tested on several image sequences. Here, an example for a broadcast application and an example for post production is presented. Both sequences include moving objects. Fig. 8 and Fig. 11 show the generated mosaics. Fig. 9, Fig. 10 and Fig. 12, Fig. 13 illustrate the results of mosaic image decomposi-

tion. Fig. 12 and Fig. 13 shows, that the automatic generation of the moving foreground label masks is sometimes not perfect. This is due to changes in illumination and the similar color of the actress's coat and the wall in the background.

Figure 8: Broadcast: Original image sequence (top) and generated mosaic (bottom). Moving objects are removed.

Figure 9: Broadcast: Edited mosaic image (top), decomposed background images (2nd row), moving object label masks (3rd row) and augmented image sequence (bottom).

7 Conclusions

A user-friendly automatic system for integrating virtual objects into image sequences is presented. The system considers occlusions of virtual objects with real moving foreground objects, whereby no 3D geometry calculation is required. The presented system is restricted to sequences taken by a rotating and zooming camera. However, this camera motion often occurs in practice. The first processing step of the proposed system is a robust and accurate estimation of camera parameters from corresponding image points. These parameters are used to generate a high-resolution mosaic, where moving objects are removed. The result is the mosaic image with the static background of the observed scene. Virtual objects can easily be added into the background mosaic image by the use of standard image editing software. Thus, the user has not to struggle with complex 3D modelling and animation tools. The image generator decomposes the edited mosaic image into an image sequence by using the estimated camera parameters and automatically calculates the required mask to position moving real objects in front of virtual objects. The advantage is that no manual compositing has to be performed by the user. The good results for real image sequences from sport transmissions and movie production have shown the practical profit.

References

[1] Lourdes de Agapito, Richard I. Hartley, and Eric Hayman. Linear self-calibration of a rotating and zooming camera. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 5–21, 1999.

[2] R. M. A. Fischler and C. Bolles. Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[3] C. Harris and M. Stephens. A combined corner and edge detector. In *4th Alvey Vision Conference*, pages 147–151, 1988.

[4] Richard Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593, 1997.

[5] Richard Hartley and Andrew Zisserman. *Multiple View Geometry*. Cambridge University Press, ISBN 0 521 62304 9, 2000.

Figure 10: Broadcast: Small sections of the original image (left) and its corresponding composed image (right).

Figure 11: Post production: Original image sequence (top) and generated mosaic (bottom). Moving objects are removed.

Figure 12: Post production: Edited mosaic image (top), decomposed background images (2nd row), moving object label masks (3rd row) and augmented image sequence (bottom).

[6] Marc Pollefeys, Reinhard Koch, and Luc Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *International Conference on Computer Vision*, pages 90–95, 1998.

[7] Yongduek Seo and Ki Sang Hong. About the self-calibration of a rotating and zooming camera: Theory an practice. In *International Conference on Computer Vision*, volume 1, pages 183–189, 1999.

[8] P. H. S. Torr and D. W. Murray. Outlier detection and motion segmentation. In P. S. Schenker, editor, *Sensor Fusion VI*, pages 432–443. SPIE volume 2059, 1993. Boston.

[9] P.H.S. Torr and D.W. Murray. The development and comparison of robust methods for estimating the fundamental matrix. *International Journal of Computer Vision*, 24(3):271–300, 1997.

Figure 13: Post production: Small sections of the original image (left) and its corresponding composed image (right).