

3-D Tracking of Shoes for Virtual Mirror Applications

P. Eisert, P. Fechteler, J. Rurainsky

Fraunhofer Institute for Telecommunications, Heinrich-Hertz-Institute
Einsteinufer 37, D-10587 Berlin, Germany

peter.eisert@hhi.fraunhofer.de

Abstract

In this paper, augmented reality techniques are used in order to create a Virtual Mirror for the real-time visualization of customized sports shoes. Similar to looking into a mirror when trying on new shoes in a shop, we create the same impression but for virtual shoes that the customer can design individually. For that purpose, we replace the real mirror by a large display that shows the mirrored input of a camera capturing the legs and shoes of a person. 3-D Tracking of both feet and exchanging the real shoes by computer graphics models gives the impression of actually wearing the virtual shoes. The 3-D motion tracker presented in this paper, exploits mainly silhouette information to achieve robust estimates for both shoes from a single camera view. The use of a hierarchical approach in an image pyramid enables real-time estimation at frame rates of more than 30 frames per second.

1. Introduction

The Virtual Mirror presented in this paper is a system for the visualization of customized shoes that a person can try on virtually and watch their appearance after having designed them individually at a terminal. This is achieved by using augmented reality techniques which combine real video with virtual objects represented by 3-D computer graphics models. The use of a mirror environment enables the augmentation of the user with artificial objects without the user being forced to wear special glasses. This is often exploited in the presentation of apparels [14, 17, 7, 16] but also shoes [15]. Other approaches also use such Virtual Mirror techniques for applications like visual effects (face distortions) [6], mobile electronic mirrors [11], or for image communication purposes [5].

Our system as shown in Fig. 1 has been created for adidas and runs in two stores, one at the Champs Elysées, Paris, the other in Lille, France. At their innovation center, a customer cannot choose only shoes from the shelf but design personalized models. Besides particular fitting to the left



Figure 1. Virtual Mirror presented at the IFA 2007 exhibition.

and right foot, the client can change the design and colors of a shoe model at a special terminal and add individual embroideries and decorations. In order to give the customer an impression how the shoes will finally look like after being manufactured, the user can step in front of the Virtual Mirror and watch him/herself wearing the new shoes.

A camera captures the customer wearing regular shoes. In order to achieve the desired mirror effect, the display outputs the horizontally flipped camera image. The display is mounted such that the shoes/legs appear at the same position, where the user would expect to see them when looking into a real mirror. In order to enhance the virtual feeling of the framework, the background is segmented and replaced by a synthetic environment. A novel 3-D motion tracker estimates the position and orientation for each foot using a model-based approach that is very robust and can easily be adapted to new shoe models. In contrast to many

existing approaches, no markers are required on the shoes. Once the exact feet positions and orientation in 3-D space are known, the computer graphics models, that have been configured and colored according to the customer's wishes, are rendered and integrated into the video stream such that the real shoes are replaced by the virtual ones. Special care has to be taken for this augmentation, since the real scene in the 2-D video should occlude parts of the virtual 3-D shoe models. Therefore, visibility for all parts of the shoe has to be computed for a given position. Since all algorithms have been implemented with real-time constraints, the customer can move freely and watch himself/herself with the new shoes that have been designed just some moments earlier.

2. System Description

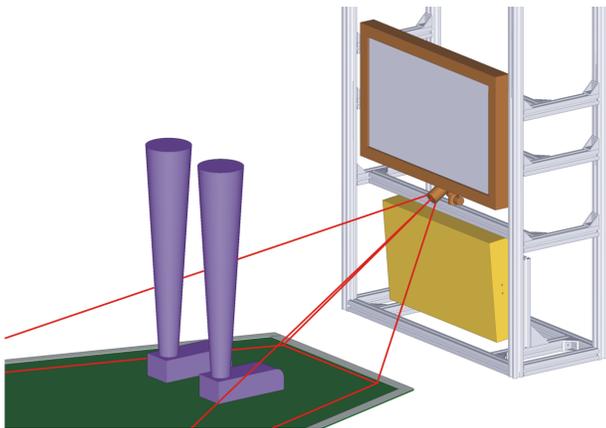


Figure 2. Architecture of the Virtual Mirror.

The system mainly consists of a single camera and a display showing the output of the Virtual Mirror. The camera is mounted close to the display and looks down, capturing the feet of a person standing in front of the system. The legs of the user are segmented from the background and displayed on the screen after mirroring the video signal horizontally. The placement of the display and the viewing direction of the camera are chosen such that an average sized person sees about the same as he/she would expect when looking in a real mirror located at the same position as the display. In order to simplify the segmentation in a real environment with changing illumination and arbitrary colors of clothes, the floor in front of the camera is painted in green to allow the use of chroma keying techniques. Moreover, an additional light below the camera reduces the effect of shadows, since the illumination in the store is mainly from the top.

All image processing algorithms, tracking, rendering and augmentation run on a single PC. It also hosts a web server that allows the control of the Virtual Mirror and interfacing with the adidas configuration and database system. In

the algorithmic flow of the Virtual Mirror, several different components have been developed like

- 2-D image pre-processing and segmentation
- gradient-based 3-D tracking
- shoe size estimation
- rendering and augmentation.

These components are described in the following sections.

3. Image Processing

The calibrated camera [8] of the Virtual Mirror continuously captures the space in front of the system and transfers the images with a resolution of 1024 by 768 pixels to the image processing components. All automatic camera control is replaced by own algorithms to avoid unexpected behavior after changes in the lighting of the scene. In order to avoid interference with artificial illumination, the shutter time is synchronized with the flickering of the lights. The camera gain is computed based on the desired mean brightness to adjust for changing illumination each time when nobody is within the range of the camera.

This idle state is determined by a change detector, that exploits information about the spatio-temporal variations in the video signal. After the camera gain has been adjusted to the current lighting situation, a background image is computed by averaging 10 consecutive video frames. This background image is used for the segmentation of the mainly green background from the shoes and the legs.



Figure 3. Background segmentation in an image pyramid.

In order to fulfill the real-time constraint, all image processing is applied in an image pyramid as illustrated in Fig. 3. The original image is filtered and down-sampled four times by a factor of two, until a final resolution of 64 by 48 pixels is reached. The segmentation algorithm starts on this lowest level by comparing all pixel colors with the corresponding ones in the background image. In the decision, whether the pixel belongs to the fore- or background,

a 3-D RGB lookup table is used having 64^3 elements. This color cube is filled adaptively with the green background pixels. In order to handle also shadows and reflections on the floor, the resulting shape of background pixels in the RGB color cube is extended by cylinder- and cone-like models. After the pixels have been classified, small holes are filled and small regions are removed until only the two legs with the shoes remain. This segmentation mask is now propagated to levels of higher resolution, where only pixels are segmented (as described above) that originate from boundary pixels in the lower level. This process is repeated until the final resolution is reached, resulting in segmentation masks for each pyramid level.

In order to determine if a person has entered the VirtualMirror, the fractions of pixels belonging to the foreground are determined along each line and column. From the vertical and horizontal distribution, different objects are detected. If both shoes are completely visible and do not occlude each other, the tracking is started. In order to restrict the search range of the tracker in the first frame, a rough guess of both shoe poses and locations are computed from the segmented shape by assuming that the feet are placed on the ground.

4. 3-D Tracking

In order to achieve robust 3-D tracking under difficult lighting conditions, we exploit mainly silhouette information, which is also used in body tracking [12, 3, 1, 4, 2] and gait analysis [18]. Sport shoes often have reflective materials with highly view dependent appearance, whereas elegant shoes might not show any texture at all. Both cases are difficult to describe by texture based tracking while the silhouette still provides enough information for tracking.

Our 3-D tracker estimates the two rigid body motion parameter sets (two times $R_x, R_y, R_z, t_x, t_y, t_z$) corresponding to the shoes from a single camera. In total, 12 parameters are estimated using an analysis-by-synthesis technique similar to the face tracker described in [10, 9]. Instead of tracking a set of distinct feature points, the entire image is exploited for robust motion estimation. 3-D computer graphics models specifying the shape of the shoes are rendered into a synthetic image approximating the camera frame. By reading out the z-buffer of the graphics card, information about the shoes' silhouettes and their dense depth information is obtained. The silhouette mask of the synthesized frame is now matched with the segmentation mask from the foreground segmentation. All motion parameters are optimized such that there is a perfect fit of real and synthetic silhouettes. However, if the shoes show sufficient texture, color information can be exploited in the analysis-by-synthesis loop exactly in the same way by additionally providing texture information to the computer graphics models.

4.1. Gradient-based 3-D Motion Estimation

The tracking process can be thought as finding the 3-D parameter set that optimally matches the 2-D silhouettes (and/or color information). However, a complete search in the 6- respectively 12-dimensional space would be very inefficient. Therefore, the parameters are directly computed using a gradient-based technique. For that purpose, the binary silhouette masks of synthetic and camera frames are first filtered using a separable 7 tap moving average filter. This operation transforms the binary object borders into linear ramps with constant gradient values. The closer a pixel is to the object, the higher the pixel values. By comparing different intensity values, information about mismatch of boundaries can be computed using the optical flow constraint equation [13]

$$\frac{\partial I(X, Y)}{\partial X} d_x + \frac{\partial I(X, Y)}{\partial Y} d_y = I(X, Y) - I'(X, Y), \quad (1)$$

where $\frac{\partial I}{\partial X}$ and $\frac{\partial I}{\partial Y}$ are the spatial derivatives at pixel position $[X \ Y]$. $I' - I$ denotes the intensity change between the filtered original and synthetic silhouette image. The 2-D displacement vector can be related with the unknown motion parameters

$$\begin{bmatrix} d_x & d_y \end{bmatrix}^T = \mathbf{d}(\Delta R_x^l, \Delta R_y^l, \Delta R_z^l, \Delta t_x^l, \Delta t_y^l, \Delta t_z^l, \Delta R_x^r, \Delta R_y^r, \Delta R_z^r, \Delta t_x^r, \Delta t_y^r, \Delta t_z^r) \quad (2)$$

using information about the rigid body motion model and the knowledge about the camera parameters given by perspective projection

$$\begin{aligned} X &= -f_x \frac{x}{z} \\ Y &= -f_y \frac{y}{z}. \end{aligned} \quad (3)$$

Here, f_x and f_y denote the scaled focal lengths that account for different pixel geometries while X and Y being the pixel indices centered around the optical axis. If the rotation of the shoe is performed around point $[x_c \ y_c \ z_c]^T$ the displacement vector (2) can be written as

$$\begin{aligned} d_x &\approx f_x \left[-\Delta R_y \left(1 - \frac{z_c}{z}\right) - \Delta R_z \left(\frac{Y}{f_y} + \frac{y_c}{z}\right) - \frac{\Delta t_x}{z} \right] + \\ &+ X \left[\Delta R_x \left(\frac{Y}{f_y} + \frac{y_c}{z}\right) - \Delta R_y \left(\frac{X}{f_x} + \frac{x_c}{z}\right) - \frac{\Delta t_z}{z} \right] \\ d_y &\approx f_y \left[\Delta R_x \left(1 - \frac{z_c}{z}\right) + \Delta R_z \left(\frac{X}{f_x} + \frac{x_c}{z}\right) - \frac{\Delta t_y}{z} \right] + \\ &+ Y \left[\Delta R_x \left(\frac{Y}{f_y} + \frac{y_c}{z}\right) - \Delta R_y \left(\frac{X}{f_x} + \frac{x_c}{z}\right) - \frac{\Delta t_z}{z} \right] \end{aligned} \quad (4)$$

Combination of (1) with (4) provides one additional equation for each pixel close to the object border. An overdetermined linear set of equations is obtained that can be

solved efficiently in a least-squares sense. Remaining errors in the motion parameter set caused by linearizations are resolved by applying the algorithm iteratively. Currently, four steps are used to converge to the desired resolution.

4.2. Pixel-based Determination of Object Properties

The algorithm for 3-D motion estimation requires knowledge about the object distance z to the camera for each pixel containing the computer graphics model. Moreover, in the scene there are two shoe models which might occlude each other or are occluded by additional objects like the trousers. Therefore, we need a way to determine whether a pixel belongs to the left or right foot, its distance to the camera, and if it is a candidate for occlusions by the trousers. All these issues can be solved efficiently by the graphics hardware. For this purpose, we render the shoe models with the current resolution used for estimation in the back buffer of the window. Reading out the z-buffer of the graphics card and converting these values to metric distances, results in a pixel-wise depth map. By coloring the triangles differently as illustrated in Fig. 4 for the left and right shoe model and the top area that might be occluded by the trousers, we can assign the pixel to one set of equations for left and right shoe motion estimation. Moreover, we can detect possible regions for outliers in the silhouette due to occluding objects.

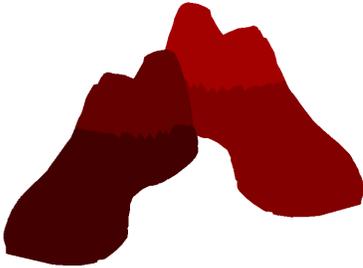


Figure 4. Color coding of different shoe parts for pixel classification.

5. Shoe Size Estimation

The Virtual Mirror should work for all different kind of shoe sizes ranging from small children's feet to extremely large models. However, different shoe sizes cannot be created by simple uniform scaling in all three dimensions, but length, width, and height vary differently when changing overall sizes. Fortunately, we can model these changes linearly for a very broad range. We define the shoe model with a size of 30 cm and scale the length with scaling factor s_z . The scaling in width s_x and height s_y has experimentally been found to be approximately

$$\begin{aligned} s_x &= 0.3674 + 0.6326 \cdot s_z \\ s_y &= 0.4518 + 0.5482 \cdot s_z. \end{aligned} \quad (5)$$

This non-uniform scaling leads to a different shape of the shoes and thus results in incorrect motion estimates if not considered correctly. Therefore, we also estimate the length of the shoes. Assuming a similar size of left and right shoe, only one scaling factor s_z has to be determined. From a single perspective view, the global scaling can usually not be estimated but here we can exploit the floor as an additional calibration hint. Usually, at least one foot is on the ground most of the time and a shoe can never be below this plane. This information can be exploited to determine the absolute size.

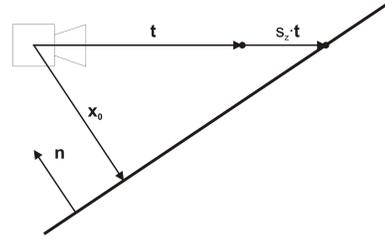


Figure 5. Coordinates for shoe size estimation.

Assume the case, where the 3-D model is too small compared to the real shoe. In this case, the motion estimator would move the shoe model closer to the camera in order to get a better fit of the silhouette. If the shoe is on the ground plane at that moment, we have an additional constraint to compute the scaling factor. We can place the shoe on the ground by scaling its translation vector \mathbf{t} by the scaling factor s_z such that the vector end on the ground plane. This is achieved by scaling the 3-D model with the same factor s_z and let the motion estimator determine the new translation vector. With the coordinate system located in the camera focal point as illustrated in Fig. 5 the scaled point $s_z \cdot \mathbf{t}$ has to lie in the plane

$$(s_z \cdot \mathbf{t} - \mathbf{x}_0) \cdot \mathbf{n} = 0, \quad (6)$$

with \mathbf{x}_0 being a point on the plane and \mathbf{n} the normal vector of the plane. Both vectors are constant for a particular setup and can be determined by camera calibration. Solving for the unknown scaling factor leads to

$$s_z = \frac{\mathbf{n} \cdot \mathbf{x}_0}{\mathbf{n} \cdot \mathbf{t}}. \quad (7)$$

This estimation only works if the shoe is placed on the ground plane. Lifting one foot leads to a larger (and incorrect) scaling factor. This is addressed by first using the smaller of the two scaling factors for the left and right foot, respectively, assuming that at least one foot touches the floor. Small jumps are still handled correctly by slowly estimating the scaling of hundreds of frames analyzing the histogram of estimated scaling factors.

6. Rendering and Augmentation

Once the 3-D motion parameters for both shoes are determined, the 3-D computer graphics models of the personalized shoes can be rendered at the correct position, such that the real shoes of the person in front of the Virtual Mirror are replaced. These 3-D models can be individually configured by a customer by selecting a base model and choosing between different sole types, materials, and colors. Additionally, individual embroideries like flags and text can be attached. From this configuration data, an individual 3-D model is composed.

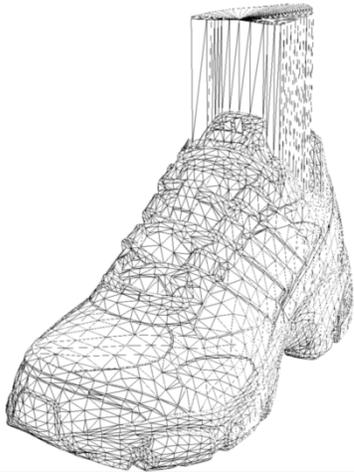


Figure 6. Hidden line representation illustrating the non-visible leg model for handling of occlusions.

For that purpose, geometry, texture, and colors of the 3-D models have to be modified to represent the chosen design. Each shoe model consists of different sub-objects composed of triangle meshes which can be replaced to create different geometries. For the modeling of different surface materials, individual texture maps are chosen from a database. Additionally, colors can be assigned to the textures in order to customize the individual parts of the shoe. This way, the customer can choose among hundreds of thousands of models and design a shoe according to personal preferences.

The left and right shoe are finally rendered using OpenGL at the position and orientation determined by the 3-D tracker. In the rendering and augmentation process, first the background is rendered at a far distance. Then, the original video is rendered using the segmentation mask as alpha channel of the RGBA texture map such that only the foreground / leg is visible. Finally, the shoes objects are overlaid, occluding the original shoes in the segmented video. However, the legs in the original 2-D video should also occlude some parts of the shoes, e.g. the interior that might be visible otherwise. By adding also a transparent non visible leg model to the scene as shown in Fig. 6, the



Figure 7. View dependent rendering for correct augmentation into video. Parts that should be later occluded by the legs (2-D video) are not visible.

z-buffer is manipulated such that all occlusions are recovered correctly and the 3-D model can be augmented into the 2-D video. Fig. 7 shows two examples of the shoe rendering with some parts, later occluded by the legs, removed.

7. Experimental Results

In this section, some results of the tracking and rendering are presented. Four different shoe models are configured and the Virtual Mirror is started. A single XGA firewire camera captures the scene at a resolution of 1024 by 768 pixels. A user enters the green space in front of the system. In all cases, the shoes are correctly detected, segmented, and tracked. Fig. 8 shows examples of the Virtual Mirror output. The upper row illustrates some frames of the original scene captured with the camera. For these frames, the corresponding results displayed on the Virtual Mirror are depicted in the lower row. It can be seen, that the 3-D computer models follow the original 3-D shoe motion correctly, even for rather extreme feet positions.

Since the entire framework should behave like a real mirror, real-time processing is required. All algorithms are therefore optimized for speed. Image processing algorithms are applied in an image pyramid and the tracking is also computed on a lower resolution level. In order to exploit multiple processor cores, a multi-threaded framework is setup with four threads for capturing/debayering, image processing, rendering, and system control. On a 2.3 GHz dual processor Pentium 4 based system, the Virtual Mirror takes about 28 ms for all computations with 4 iterations of the tracker, leading to a maximum frame rate of about 35 Hz.

8. Conclusions

We have presented a system for the real-time 3-D tracking of shoes in a Virtual Mirror environment. From a single camera the rigid body motion of left and right foot are estimated using linear and low-complexity optimization methods. The tracking algorithm is not restricted to shoe models, but can also be applied to other objects if



Figure 8. **Upper row:** Scene captured with the Virtual Mirror camera. **Lower row:** Virtual Mirror output augmented with customized shoes.

a 3-D geometry description is available. The motion information is then used to render new personalized sports shoes into the real scene such that the person can watch himself/herself wearing the new shoes. The system is accessible to the public in stores in Paris and Lille and new installations with updated features are also planned in the next future.

Acknowledgements

The work presented in this paper has been developed with the support of the European Network of Excellence VISNET II (Contract IST-1-038398).

References

- [1] A. Agarwal and B. Triggs. 3D human pose from silhouettes by relevance vector regression. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, Washington, USA, June 2004.
- [2] A. Balan, L. Sigal, M. Black, J. Davis, and H. Haussecker. Detailed human shape and pose from images. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, Minneapolis, USA, June 2007.
- [3] K. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *Proc. Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 77–84, June 2003.
- [4] K. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette across time part ii: Applications to human modeling and markerless motion tracking. *International Journal of Computer Vision*, Jul. 2005.
- [5] C. Cullinan and S. Agamanolis. Reflexion: A responsive virtual mirror for interpersonal communication. In *Proc. ECSCW 2003 8th European Conference on Computer Supported Cooperative Work*, Helsinki, Finland, Sep. 2003.
- [6] T. Darrell, G. Gordon, J. Woodfill, and M. Harville. A virtual mirror interface using real-time robust face tracking. In *Proc. Third International Conference on Face and Gesture Recognition*, Nara, Japan, Apr. 1998.
- [7] J. Ehara and H. Saito. Texture overlay for virtual clothing based on PCA of silhouettes. In *Proc. Int. Symposium on Mixed and Augmented Reality ISMAR 2006*, pages 139–142, Santa Barbara, USA, Oct. 2006.
- [8] P. Eisert. Model-based camera calibration using analysis by synthesis techniques. In *Proc. International Workshop on Vision, Modeling, and Visualization*, pages 307–314, Erlangen, Germany, Nov. 2002.
- [9] P. Eisert. MPEG-4 facial animation in video analysis and synthesis. *International Journal of Imaging Systems and Technology*, 13(5):245–256, Mar. 2003. invited paper.
- [10] P. Eisert and B. Girod. Analyzing facial expressions for virtual conferencing. *IEEE Computer Graphics and Applications*, 18(5):70–78, Sep. 1998.
- [11] A. Francois and E. Kang. A handheld mirror simulation. In *Proc. International Conference on Multimedia and Expo (ICME)*, volume 2, pages 745–748, Los Angeles, USA, Jul. 2003.
- [12] A. Hilton, D. Beresford, T. Gentils, R. Smith, W. Sun, and J. Illingworth. Whole-body modelling of people from multi-view images to populate virtual worlds. *Visual Computer: International Journal of Computer Graphics*, 16(7):411–436, 2000.
- [13] B. K. P. Horn. *Robot Vision*. MIT Press, Cambridge, 1986.
- [14] J. Hoshino and H. Saito. A match moving technique for merging CG and human video sequences. In *Proc. International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Salt lake City, USA, May 2001.
- [15] S. Mottura, L. Greci, E. Travaini, G. Vigano, and M. Sacco. Mag-icmirror & footglove: A new system for the customized shoe try-on. In *Proc. 17th CIRP Design Conference*, Berlin, Germany, Mar. 2007.
- [16] V. Scholz and M. Magnor. Texture replacement of garments in monocular video sequences. In *Proc. of the 17th Eurographics Symposium on Rendering EGSR 2006*, pages 305–312, Nicosia, Cyprus, June 2006.
- [17] A. Taguchi, T. Aoki, and H. Yasuda. A study on real-time virtual clothing system based on two-dimensional plane model. In *Proc. 6th Symposium on Information and Telecommunication Technologies APSITT 2005*, pages 126–130, Yangon, Myanmar, Nov. 2005.
- [18] G. Veres, L. Gordon, J. Carter, and M. Nixon. What image information is important in silhouette-based gait recognition? In *Proc. Computer Vision and Pattern Recognition (CVPR)*, Washington, USA, June 2004.