Proceedings of EuroHaptics 2004, Munich Germany, June 5-7, 2004.

Flexible and Configurable Haptic Environments through Virtual Graphical Force Sensors

Michael F. Zaeh[1], Stella M. Clarke[1]

[1] Institute for Machine Tools and Industrial Management
Technische Universitaet Muenchen
Boltzmannstrasse 15
85748 Garching, Germany
{michael.zaeh, stella.clarke}@iwb.tum.de
http://www.iwb.tum.de

Abstract. This paper presents a solution to virtual sensors which enables users to create virtual haptic environments in a flexible, easy to use and time efficient manner. The usefulness of virtual sensors has been shown in a number of applications, such as telepresent micro-assembly, in which haptic feedback is often desired without mechanical impact with a "real" sensor. Instead of requiring each virtual sensor configuration to be individually programmed, this paper presents a method to graphically generate virtual haptic environments with user-friendly graphical software such as bitmap and VRML editors. Both two-dimensional and three-dimensional implementations are discussed.

1 Introduction

Haptic feedback plays a crucial role in telerobotic operations, in which the goal is to provide a human operator with a remote, yet intuitive, operating environment. The effectiveness of haptic feedback in enhancing user intuition has been proved under many circumstances. For example, force feedback in an Argonne E2 master-slave system was shown to reduce task completion times by almost 50% [1]. In telepresence scenarios, haptic feedback is often desired without necessary contact with a "real" sensor, such as a force or contact sensor. Such "virtual" feedback can be obtained from so-called "virtual sensors".

The output from virtual sensors is usually inferred from the signals of other system-relevant sensors or information. For example, [2] presents a virtual sensor in which pressure values in a spark-ignited car motor are inferred from ion current measurements in the spark plug. In [3], force feedback values from the cantilevered tip of an atomic force microscope are inferred from contact mechanics.

In many telepresence scenarios, the aim of virtual sensors is to increase operator intuition. For example, to provide the operator with a "familiar" operating environment, the micro-assembly system presented in [4] uses a virtual sensor to simulate gravity at the micro scale, where gravity usually plays a relatively insignificant role.

In this paper, the virtual sensor approach developed infers force feedback values from position values in telepresence scenarios. Moreover, the mapping of position values to force values is carried out in a graphical manner. This means that a graphical representation of the teleoperator environment is modified to include virtual shapes, which are drawn in at positions where virtual force feedback is desired. In two-dimensional teleoperations, a 2D bitmap image is modified, and in three-dimensional operations, a VRML world is used. Information regarding magnitudes and directions of force feedback values are stored within the individual red, green and blue (RGB) color intensities of the virtual shapes. Color gradients can be drawn in to gradually exert virtual forces or to steadily change force directions. This proposed virtual graphical force sensor (abbreviated VGFS hereon), is shown to provide a flexible solution to virtual sensors in general.

2 Two-dimensional Implementation

The following outlined two-dimensional graphical force sensor was implemented within a game-oriented telepresence scenario. The task is to steer a video camera through a two-dimensional labyrinth using a haptic feedback joystick. The operator feels a resisting force through the haptic joystick when a wall is touched. Optimally we desire the operator to be warned of a possible collision before contact with the wall is made; hence a virtual force sensor was decided upon.

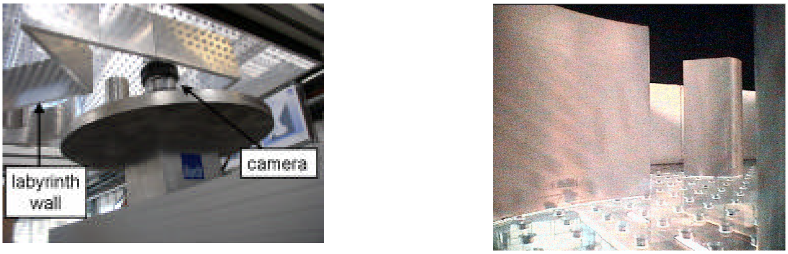

Fig. 1. (a) View of camera and labyrinth from below (b) View from camera inside labyrinth

The bitmap file containing the virtual force information is depicted within the flow chart of Fig. 2. One pixel in the bitmap represents one square millimeter in the actual labyrinth. The flow chart shows how the VGFS reads in the current axes positions ($x_{current}$, $y_{current}$) to determine where the camera currently is within the labyrinth. The VGFS then accesses the bitmap and queries which color lies at the corresponding ($x_{current}$, $y_{current}$) of the bitmap. Once the color is retrieved, the red, green and blue components are extracted. Each color intensity ranges from 0 to 254 (the value 255 is ignored to guarantee a middle value, 127, in which zero force is exerted).

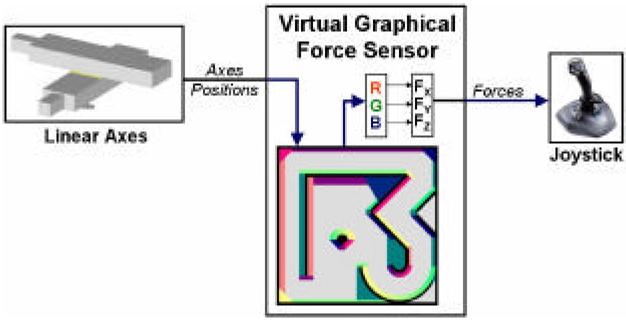

Fig. 2. Two-dimensional VGFS implementation overview. The VGFS program uses the current axes positions to find the corresponding location in the bitmap image, determine which color lies there, and exert the mapped forces on the joystick

The VGFS was implemented in C++, utilizing a National Instruments library to read axes positions through a motion controller card, and a graphics library to extrapolate color information from the bitmap image. Microsoft's MFC libraries were used to create a simple GUI. Although not relevant to the VGFS, a UDP communication layer was incorporated between the VGFS and the receiving joystick application, making this a true telepresence scenario.

3 Three-dimensional Implementation

Although three-dimensional bitmaps are programmatically possible, they have relatively large file sizes and there exists no easy-to-use graphical editors with which to create the virtual 3D haptic fields. Instead, a VRML file (*.wrl) was utilized.

This implemented micro-assembly telerobot scenario consists of a static tool head and moving assembly tray.

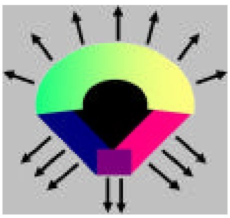

Fig. 3. 2D color-force mapping showing the use of gradients for directional change

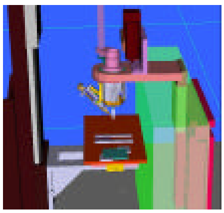

Fig. 4. 3D VRML representation with virtual box towards the right to constrain motion of assembly tray

The capability for directional information to be stored in virtual sensors provides possibilities beyond specifying force directions. It can also be used for position guidance. For example, the virtual block in Fig. 4 can not only be used to exert a force to the left on the assembly tray; it can also be used to block movement through the sensor from the left-hand side, but allow movement of the assembly tray from the right-hand side.

4 Computational Performance

In the two-dimensional case, because bitmaps are essentially uncompressed images which are accessed once every 50 milliseconds in the VGFS, it is important to check the process and memory usage. The resolution of the bitmap image employed for the labyrinth was higher than necessary, with dimensions of 538 x 549 pixels, and a size of 866 kilobytes. The employed computer lies at the low end of today's computer standards, with a Pentium 4 1.80GHz processor, 512MB main memory, 512kB level 2 cache, and 8kB level 1 cache. The bitmap image fits into the main memory with ease. In addition, considering that subsequent pixel accesses will probably occur close together, a significant amount can be stored in the level 2 cache, thereby possessing access rates close to the processor clock speed.

In three dimensions, the equivalent size of the same bitmap image would be approximately 25MB, which is too large to be efficiently accessed 20 times per second. A real-time evaluation of the VRML setup was not undertaken, but research suggests possible positive results. For example, [5] explains how collision detection of complex interweaving pipeline shapes, each consisting of 140,000 polygons, could be carried out at a rate of 250 times per second with a 90MHz CPU.

Acknowledgements

This work was supported by the German Research Foundation (DFG) within the SFB 453 project on "High-Fidelity Telepresence and Teleaction". Thanks also to Mike Ehrenstrasser and Johannes Schilp for the images of their micro-assembly robot.

References

1. Massimino, M.J, Sheridan, T.B., Roseborough, J.B.: One Handed Tracking in Six Degrees of Freedom. In: Proceedings of IEEE International Conference on Systems, Man and Cybernetics (1989) 498-503
2. Holmberg, U., Hellring, M.: A Simple Virtual Sensor for Combustion Timing. In: Journal of Dynamic Systems, Measurement, and Control 125 (2003)
3. Sitti, M., Hashimoto, H.: Tele-Touch Feedback of Surfaces at the Nano Scale: Modeling and Experiments. In: IEEE/ASME Transactions on Mechatronics (2002)
4. Schilp, J., Ehrenstrasser, M., Clarke, S., Petzold, B., Zaeh, M.F.: Smart Sensor Application in Teleoperated Micro-Assembly Systems. In: Proceedings of SPIE Vol. #5263A (2003)
5. Lin, M., Manocha, D., Cohen, J., Gottschalk, S.: Collision Detection: Algorithms and Applications. In: Proceedings of the 1996 Workshop on the Algorithmic Foundations of Robots (1996) 129-142