Annotating Real World objects using a Steerable Projector-Camera Unit

Andreas Butz

Media Informatics
University of Munich
Munich, Germany

Christian Schmitz

Computer Science
Saarland University
Saarbrücken, Germany

Abstract

We describe the annotation of real world objects with labels similar to the "tool tips" known from desktop interfaces using a steerable projector-camera unit. Labels containing text or graphics are projected onto objects or next to objects on suitable surfaces and are placed automatically according to a set of weighted criteria ensuring an unambiguous mapping between labels and labeled objects as well as consistency among the set of labels. The method has been implemented in our instrumented environment to form the basis for making this environment – to some extent – self-explaining.

1. Introduction

Instrumented environments are one way of exploring ubiquitous computing scenarios within limited areas of space. Instrumentation can be achieved by various kinds of sensors and actuators, including conventional displays, cameras and projectors. The SUPIE (Saarland University Pervasive Instrumented Environment) is one example of such an instrumented environment in the size of a regular office, and one of its central elements is a steerable projector-camera unit mounted on the ceiling in the middle of the room (see figure 1). One core idea is that the steerable projector-camera unit turns the whole room into a large display continuum, where other displays are consistently integrated as areas of higher resolution and enhanced interactivity. A conceptual model for this was presented in [3]. The steerable projector might eventually be substituted by electronic wall papers or carpeting with display capabilities, and we try to develop interaction concepts which will then still remain usable. One way to make an instrumented environment interactive is to visually create interactive elements in it. A straightforward approach is to transfer concepts from our current direct manipulation or WIMP (Windows, Icons, Menus, Pointing device) interfaces to such an environment. In our earlier work, we have already presented a search functionality for physical objects [4] which uses simple spots of light to highlight physical objects using a steerable projector.

Figure 1: The steerable projector used in our environment

Another important aspect of WIMP GUIs is their explorability. This means, that a novice user can explore the interface with her pointing device and the different parts of the interface will reveal their functionality. One way to create explorability are the so-called "tool tips", which are small labels automatically appearing when the pointer rests on a particular item for a short time without triggering an action. They thereby show the function of a particular widget before it is operated. In an instrumented environment, these tool tips can also be applied to physical objects. While adding a label to an already projected widget is trivial, labeling a potentially complex 3D object in an unambiguous way is a complex task in itself. After briefly reviewing related work in this field, we will describe our algorithm for placing projected annotations and present some examples of its results.

2. Related Work

Label placement and annotation of graphics have a long tradition in manually produced graphics. Maps and instruction

manuals are unthinkable without them. With the advent of more powerful techniques for automated layout and design of communicative graphics, research has started to investigate the automated placement of text and other annotations in 2D and 3D graphics [7, 6].

Most inspiration for our label placement was taken from earlier work on annotation of 3D objects [7, 6]. Initial work on steerable projectors already highlights their capability to create ubiquitous displays[5] and shows examples how these displays can be used to annotate physical objects. Consistency between several annotations and automatic placement, however, are not considered. More recent work in this field shows how a steerable projector-camera unit can be used to automatically determine and track display surfaces[2]. In our work we do not emphasize this aspect, but the result of this process of display surface determination could easily be used to supply the set of potential display surfaces which we currently define manually.

Recent work in Augmented Reality investigates the annotation of physical and virtual objects for users wearing an optical see-through head worn display[1], but the emphasis there is on maintaining visual continuity across frames in a real time rendering situation. Our scenario is less challenging in terms of speed requirements, which makes it possible to use more sophisticated methods for initial placement of labels. While the labels displayed in a head-worn display always live in the same projection plane, physically projected labels also have to account for the available projection surfaces in the environment, their relative orientation and physical properties.

3. Annotating Physical Objects

Our method for the annotation of physical objects uses a steerable projector-camera unit (figure 1) to project rectified images onto every planar surface in the room. Different surfaces in a room have different qualities and some are more appropriate for projection than others. The best projections are obtained on bright, planar and reflecting surfaces, whereas dark, specular and non-planar surfaces usually don't offer good conditions for projections. In office environments, such as our instrumented test environment, there are many possibilities for projection, as most walls, ceilings and tables are white and planar.

In order to annotate a physical object, our algorithm uses information about the object's geometry and available display surfaces in its proximity. Currently, physical objects and possible display surfaces are defined manually, but at least the display surfaces could also be derived by a process similar to the display area detection described in [2]. Physical objects are described by the coordinates of their bounding box. A physical object can also consist of multiple subparts, which are modeled in a part-of hierarchy, and for each

subpart, the bounding box is again represented. Labels have a certain freedom regarding their actual placement around or on the object. For each possible display surface, a quality factor is given, which rates the expected projection quality on it. Surfaces can be either public or private. Public here means that they can be used for all objects, whereas private surfaces are mostly defined on the objects themselves and should only be used for exactly these objects. The set of possible display surfaces is the basis for determining the best place for an annotation as shown in figure 2.
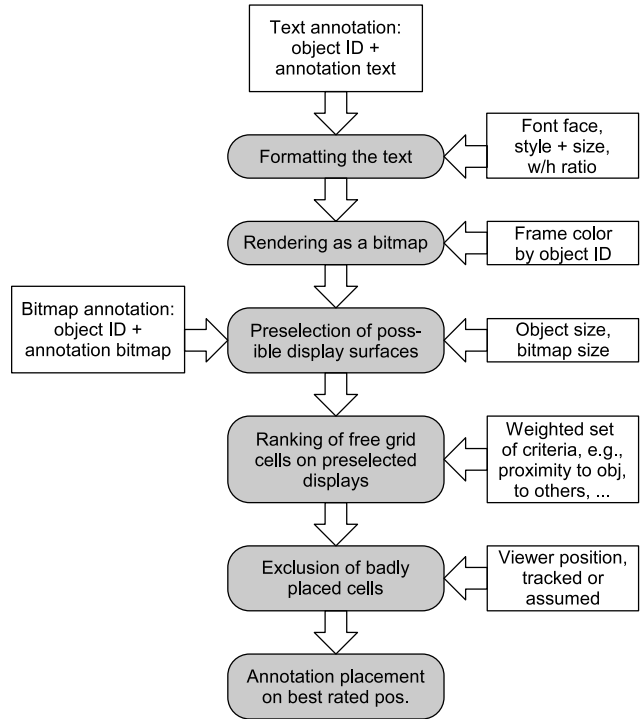

Figure 2: Placement process for annotations

As the surfaces are modeled with maximum sizes, usually several annotations can be placed onto one single surface, and they can also contain widgets. The algorithm keeps track, which parts of the surfaces are currently in use and which are free. This is achieved by dividing the surface into a grid. The cell size of the grid is currently set to $1cm^2$ in the physical world, which roughly corresponds to the mechanical precision of our steerable projector.

An annotation can be either a string or a picture. Text annotations are automatically wrapped to a width to height ratio of about 2:1, which in most cases makes them well readable. In the first step a pre-selection of all public display surfaces is done. Display surfaces which are too far away from the object (in relation to the object's and the annotation's size) are excluded. The set of remaining display surfaces including the private displays of the reference

object and its super-objects in the part-of hierarchy is then evaluated according to a weighted set of criteria. The most obvious criterion is the distance of the display surface to the annotated (or reference) object. Near surfaces are rated higher than far ones. Making an annotation non-ambiguous means that it should be visibly closer to its reference object than to all other objects.

Figure 3: Annotation of a complex real world object

Multiple annotations with different reference objects are not placed close to each other. Hence the distance between annotations influences the evaluation of a display surface, too. The position of the user plays an important role, because projections in the real world always depend on the view point. When a new annotation is added, it is placed close to the line of sight from the user to the object so that it can be clearly attributed to its reference object. At present, our algorithm only takes into account the position of a single user. Another criterion is the position of the annotation relative to the reference object. For each object, a preference can be specified whether annotations should be placed to the left or right, in front or behind, and below or above it. These preferences, together with the expected display quality (see above) also influence the rating.

As some of the criteria are more important than others, priorities are specified by assigning weights to the criteria. The evaluation of all cells of the display surfaces according to the specified set of criteria finally results in a ranking. In the next step the cells with the highest ranking are checked for additional criteria: The user must not occlude them and look at them in a sufficiently steep angle. For lack of a decent body tracking system, we currently assume the user position to be the middle of the room, but a tracked position will be handled correctly when available.

Finally, there must be enough free cells in the direct neighborhood of the selected cell, so that the annotation can be placed there. If this is the case, the annotation is added and the involved cells are blocked for further annotations. Otherwise, the next cell in the ranking is checked until a feasible placement is found. The system also allows the removal of existing annotations. If an annotation is removed, the status of the involved cells is changed back to "free".

With the described mechanism annotations can be added to and removed from all physical objects in the environment, which are represented in the 3D model. Annotations with the same reference object are marked with an equally colored border, while annotations for different objects have different border colors. This enhances the visual coherency and, in the case of multiple annotations between objects, makes clear which annotations belong to the same object.

We have implemented our approach within the SUPIE (Saarland University Pervasive Instrumented Environment). Our demonstrator is written in Java and integrated into the lab's software infrastructure, which allows, for example, shared access to the steerable projector. We have used this approach to automatically label objects in the room and on a shelf when user activity was detected nearby. Data about physical objects and display surfaces is currently stored in XML files for easy extension of the system. As a test case we have labeled the subcomponents of a one meter tall model of a Swiss Army Knife (Figure 3).

4. Conclusions and Future Work

In this paper we have presented a method for the automatic annotation of physical objects in an instrumented environment using a steerable projector-camera unit. It can be used to make the instrumented environment explorable and self-explaining by automatically labeling physical objects in it.

Currently, the set of possible display surfaces is defined manually. This process could be partially automated by methods similar to the ones described in [2]. Their result could be stored in the current display surface definition file and our system could then be used without change. One limitation of using automatic surface detection with our method is that it won't yield the information whether a surface should be considered "private" to a certain object, unless it can determine whether this surface is part of the object's actual geometry.

Consequently, exact object acquisition via stereo vision or structured light would also be a desirable feature. On

the other hand, the fact, that annotations are placed, already means that there is a model with a certain semantic of the environment, which requires manual modeling anyway.

If the objects to be labeled are highly mobile, yet another approach would be to use our searchlight[4] work to determine actual object positions. If the object's actual geometry is adequately modeled, the 3D scene could be updated accordingly and our annotation method would work as proposed in this paper.

Acknowledgments

References

[1] Blaine Bell, Steven Feiner, and Tobias Höllerer. View management for virtual and augmented reality. In *Proceedings of UIST 2001*, pages 101–110, San Diego, CA, November 11-14 2001. ACM press.

[2] Stanislaw Borkowski, Olivier Riff, and James L. Crowley. Projecting rectified images in an augmented environment. In *IEEE International Workshop on Projector-Camera Systems*, Nice Acropolis, Nice, France, October 12 2003. IEEE.

[3] Andreas Butz and Antonio Krüger. A generalized peephole metaphor for augmented reality and instrumented environments. In *Proceedings of The International Workshop on Software Technology for Augmented Reality Systems (STARS), Oct. 7, 2003 in Tokyo, Japan*, 2003.

[4] Andreas Butz, Michael Schneider, and Mira Spassova. Searchlight - a lightweight search function for pervasive environments. In *Proceedings of Pervasive 2004*, LNCS. Springer, 2004.

[5] Claudio Pinhanez. Using a steerable projector and a camera to transform surfaces into interactive displays. In *Proc. of CHI'01 - Short Talks*, Seattle, April 2001.

[6] Bernhard Preim, Andreas Raab, and Thomas Strothotte. Coherent zooming of illustrations with 3d-graphics and textual labels. In *Proc. of Graphics Interface*, 1997.

[7] Thomas Rist, Antonio Krüger, Georg Schneider, and Detlev Zimmermann. AWI: A workbench for semi-automated illustration design. In *Proc. AVI '94, Bari, Italy*, 1994.

---

[1] Flexible User Interfaces for Distributed Ubiquitous Machinery