Vision-based sensor fusion for Human-Computer Interaction

Sébastien Grange[1], Emilio Casanova[1], Terrence Fong[2], and Charles Baur[1]

[1]*Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland*
{emilio.casanova | sebastien.grange | charles.baur} @ epfl.ch
[2]*The Robotics Institute, Carnegie Mellon University, terry@cs.cmu.edu*

Abstract

This paper describes the development of efficient computer vision techniques for human-computer interaction. Our approach combines range and color information to achieve efficient, robust tracking using consumer-level computer hardware and cameras. In this paper, we present the design of the Human Oriented Tracking (HOT) library, present our initial results, and describe our current efforts to improve HOT's performance through model-based tracking.

1. Introduction

The past decade has seen an exponential improvement in processing capability and performance. Computers are increasingly becoming ubiquitous, indispensible tools in our lives. Our future will likely be populated with a range of devices, both personal and public, that perform tasks and services, both on demand and automatically.

To gain the maximum benefit from these tools, we need to develop richer, more capable interface techniques than currently exist. We need to develop methods that enable humans and computers to communicate naturally. We need to move beyond cumbersome input hardware (keyboards, mice, etc.) and screen-based displays. In short, we need to develop effective, natural, and above all, transparent methods for human-computer interaction (HCI).

To address this need, many researchers are now developing computer vision-based interfaces. A key advantage of these systems over traditional interfaces is that the "interaction" can be entirely passive. That is, vision enables computers to perceive the user, to classify his movements and activities, and to react accordingly.

This paper describes the Human Oriented Tracking (HOT) library. We developed HOT as a tool for building vision-based interfaces. The design centers on a sensor-fusion based tracker that can efficiently detect, segment, and follow human features (head, hands, etc). Moreover, HOT is designed to provide good performance using consumer-level computer hardware and cameras.

We have used HOT to build a variety of applications, including a mobile robot teleoperation interface and a virtual whiteboard. Currently, we are working to improve the HOT architecture by exploiting spatial constraints derived from hierarchical, shape-based models. Our preliminary results indicate that this sensor fusion and model-based tracking approach will enable HOT to deal with a wide range of complex objects and scenes.

2. Related Research

2.1 Vision-based interfaces

A great deal of work has been performed in the field of human tracking, particularly for video-based surveillance applications [3][6][11].

In [4], a combination of range data, color data and face pattern recognition is used to track humans. This system can track multiple users and locates their heads. The sensor fusion scheme is reported to work well, even in crowded environments, and with remarkable accuracy. However, the system requires three computers and dedicated hardware, training of the neural network, and only tracks head position. The main difference with our work is that our system runs on standard PC hardware and provides more detailed information about the human posture and gestures.

In [1], a system is presented that builds and tracks a blob-based model of the human body. The model is then used to interact with virtual characters. This system is based on adaptive background subtraction and is thus limited:

- it only tolerates one person in the image
- it does not differentiate people from objects
- a static background is required and only a fixed camera can be used

2.2 Model tracking

Some researchers have applied complex statistical models to a disparity map in order to register a model on live video. In [2], a disparity map is used to extract blobs, which are then statistically mapped onto a predefined, articulated structure. Range data allows the system to deal with occlusions better than 2-D based

trackers. However, because of its sensitivity to initialization and the fact that it only uses intensity image combined with range data, this tracker can currently only run under limited conditions. Another approach is to model each target segment of a rigid model as a planar patch bounded by the convex hull of two circles, and to use both edge and region information to match the model to the target [7]. The difference between our work and these other systems is that we combine color and stereo vision to achieve better and faster tracking.

3. HOT

3.1 Overview

The Human Oriented Tracking (HOT) library is a layered architecture for active interfaces that provides a robust feature tracker, geometric and dynamic feature modeling, and parametric activity monitoring. The HOT architecture is presented in Figure 1.

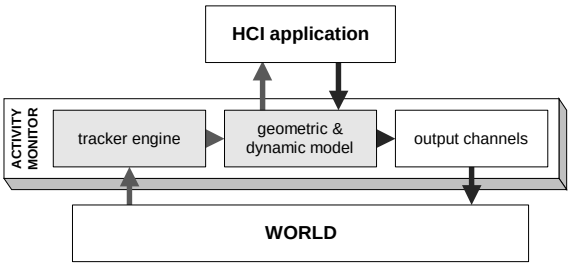

Figure 1: *HOT architecture*

HOT's feature tracker combines normalized color filtering with stereo vision. Normalized color provides fast 2D object localization, while stereo provides shape, size and range measurements. HOT's modeling layer processes the tracker output using a Kalman filter to build a geometric model and to segment feature motion.

The human model extracted by HOT is then processed to obtain interaction parameters for use by HCI applications.

3.2 Design

HOT contains three distinct parts, namely a feature tracker, a model matcher and a model interpreter. These parts are designed to operate largely independent of one another (Figure 2).

The feature tracker is where range and normalized color modalities are combined to perform robust, fast feature tracking. We compute disparity maps with the SRI Small Vision System [5]. Table I describes the properties that each modality brings to the tracker.

While neither stereo nor color by itself is sufficient to perform reliable detection and tracking, even a simplistic combination of the two can produce good results (Figure 3).

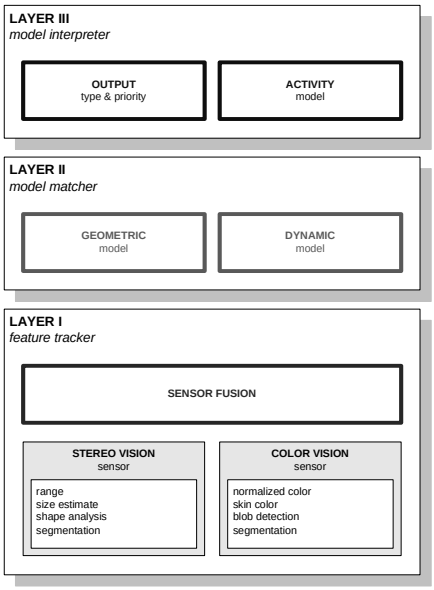

Figure 2: *HOT integration layers*

Table I: *modalities and their respective properties*

feature	stereo	color
depth estimation	+	-
skin detection	-	+
sensitivity to texture	-	+
sensitivity to light condition	+	-
real-world size estimate	+	-
shape analysis	+	+
identification	-	+

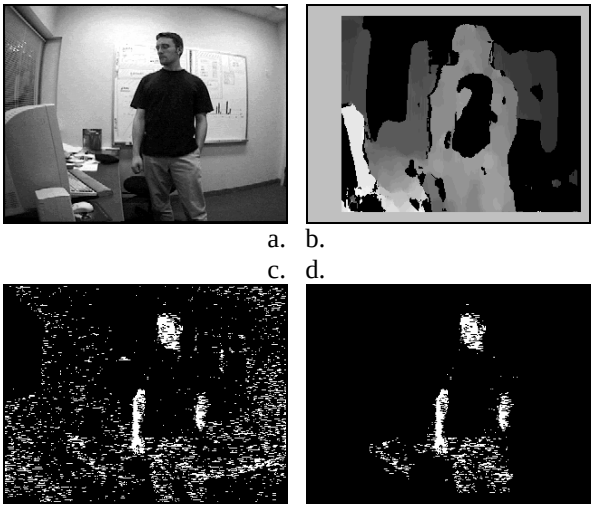

Figure 3: *Color and range image filtering (a. original image, b. disparity map, c. normalized skin-color filter, d. combined filter)*

In HOT, range and color information are used both for feature detection and for tracking. In the detection phase, we look for a blob that displays a given color and is of a

known real-world size. After filtering the image for the feature color and computing the disparity map, a histogram of color-filtered pixels is built with respect to disparity. The filtered disparity map is then decomposed into layers containing possible candidates for the target feature based on the real-world area contained in the layer. Each feature is then evaluated and the most likely match is retained.

In the tracking phase, we use a simple Kalman filter to predict the feature position in the image. At each frame, the search area is filtered in disparity and in color using the depth and color values from the previous frame. A simple binary correlation is then performed to find the best match. The overlapping area between the feature at the previous frame and the newly found feature gives a measure of the tracking performance (Figure 4).

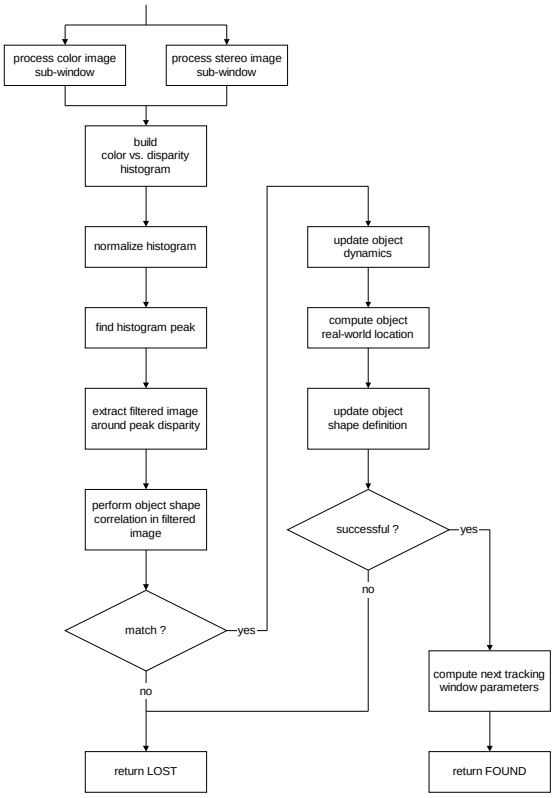

Figure 4: *feature tracking using sensor fusion*

We can use this detection and tracking strategy to find the head and hands of a human being. In this case, *a priori* information consists of a normalized skin-color locus and the respective size (in cm) of each feature.

The most significant benefits of fusing range and color modalities are robustness and performance:

- combining a depth filter with a color filter leads to better segmentation, while allowing each filter to be more tolerant. This removes the need for adaptive filters.
- the normalized color signature of an object combined with its real-world size are strong cues that prevent false position detection.

The model matcher matches the features detected during the first stage into a simplistic human model (Figure 5). The model computes spatial parameters from the human's head and hands position, as well as motion vectors extracted from the human gestures. Simple geometric consistency checks are then performed to ensure that the model is valid.

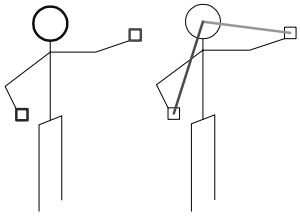

Figure 5: *Human model used to perform activity monitoring*

The model interpreter computes parameters that HCI applications can use in conjunction with the model data to interpret the human actions. These *a priori* parameters include a history of hands, arms and body "activity" measures that can be used to classify activities ("the person is walking away", "the person is moving something on that table", etc.)

3.3. Results

Using a Pentium III 500 MHz processor and two inexpensive analog cameras and digitizers, our system detects humans and performs head tracking at 25 Hz. Head localization is sufficiently accurate to determine where people are in the room and if they are standing or sitting.

Figure 6: *Examples of human features extraction and human gesture extraction*

When hand and gesture tracking is enabled, the performance decreases to about 19 Hz. Hand tracking is less reliable than head tracking, mostly due to fast and frequent occlusions and shape changes of a human hand. Figure 6 shows some examples of HOT detection and tracking capabilities (the red/green markers respectively indicate the right/left hand, vectors indicate segmented gestures).

Figure 7: *Visual gesturing for vehicle teleoperation*

As a case study, we used HOT as an input modality for mobile robot control [10]. The GestureDriver system translated hand positions and/or gestures into robot motion commands. When we initially tested the system, we found that users had difficulty controlling the robot. Analysis revealed that this was due to the localization accuracy of the HOT tracker. Specifically, the stereo method provides fairly coarse range maps and is somewhat noisy (even under constant illumination). However, once users were familiarized with the tracker's limitations, they quickly learned to accurately position the robot. Figure 7 shows a user driving a robot using virtual joystick gestures.

4. Integrating model and tracker

While the HOT feature tracker proved to be robust and fast, the system as a whole has limitations, notably:

- it is quite sensitive to significant changes in lighting conditions
- it does not treat the human model as a whole, but rather makes up a model from independent features that are "likely" to be head and hands

Thus, we are now developing a solution to this problem. Instead of deriving a human model from independent features, the new tracker algorithm in HOT will include constraints from a pre-defined, geometric model to better deal with environment changes and occlusion.

4.1 Overview

The new tracker architecture can be applied to the tracking of any rigid deformable structure. The system benefits from the robustness of the sensor fusion scheme developed in HOT, but takes additional information from a primitive-based model to dynamically redefine each feature's geometry and color boundaries during tracking.

This architecture has two components: a model definition interface, which lets the user define an arbitrary model on any object using ellipse primitives, and a dynamic algorithm that combines feature detection (using model constraints) and feature tracking (using color and range information).

4.2 Design

The new tracking architecture, called MBOT (for Model-Based Object Tracking) uses ellipse primitives (Figure 8).

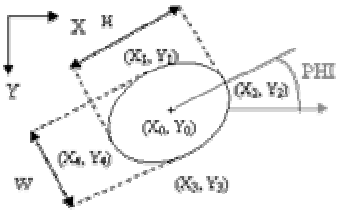

Figure 8: *Primitive used for model definition.*

Model definition is performed via a graphical user interface. The user defines a hierarchical model for any articulated object using a set of ellipses connected to one another. The relative movement between ellipses is constrained (Figure 9).

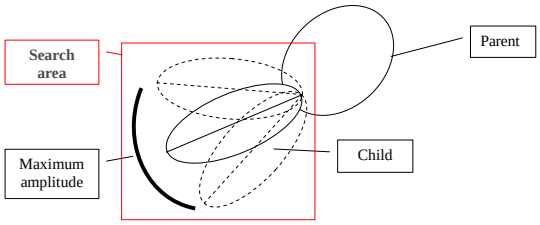

Figure 9: Link between primitive ellipses

Figure 10 shows an example of such a primitive-based model for a human being.

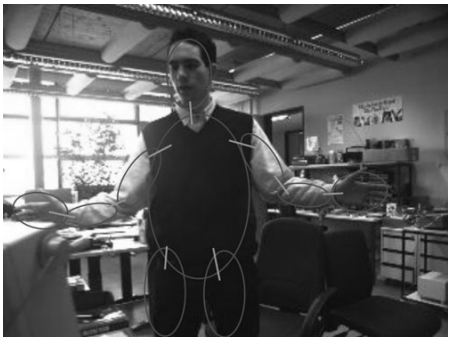

Figure 10: *primitive-based model of a human being*

While ellipses and models based on simple primitives have often been used in the past, our approach is different in that it defines the ellipses as real-world elements having real-world dimensions. Specifically, the model is composed of tridimensional planar ellipses that can match rigid elements of any size and shape.

An initial match is performed once the object has been defined. To locate each object component, we apply a loose disparity and normalized color filter to the image pair. A list of feature candidates are then segmented using a recursive, connected-compound labeling algorithm. Each candidate is then matched to a 3D planar ellipse. We use the projected real-world area of each feature candidate as a discriminant to identify the best match (Figure 11).

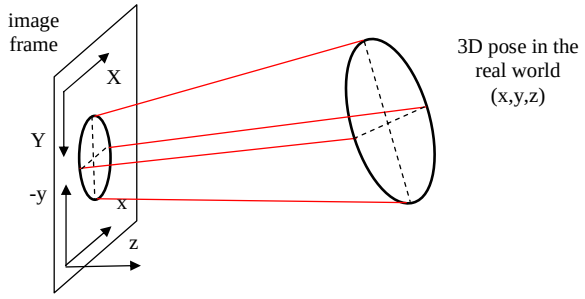

Figure 11: *Real-world projected area as a discriminant*

Tracking is performed once at least one feature from the model has been identified. The algorithm looks for the missing parents and children ellipses of the known features using information from the model constraints (relative position, relative disparity, real-world feature size, likely normalized color). Once a feature is identified, a local value for its dominant normalized color and its average depth is extracted. These local values are used in the same sensor fusion tracking scheme that was used in HOT, until the feature is lost (due to occlusion, light condition changes or dramatic shape changes). The model is constantly checked for consistency, and any feature that does not fit is considered lost. All lost features are extrapolated until they are found again.

4.3 Preliminary results

Our current tracker demonstrates the robustness of the method. Figure 12 shows the initialization phase of a simple articulated object used for testing purposes.

As can shown in figure 13, the tracker successfully tracks all the features. If one of the two features is lost, the constraints from the model strongly limit the search area based on the last observed feature position. The tracker then redefines a color, shape and disparity value for the lost feature and returns to the fast tracking algorithm.

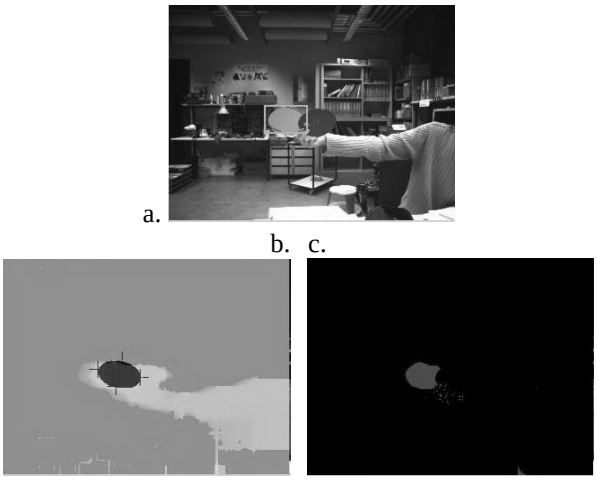

Figure 12: *Model building and initialization*
(a. original image and feature definition,
b. and c. automatic stereo and color segmentation)

Figure 13: *Tracking of an object made of two primitives*

5. Future work

Our tracking strategy works well for simple articulated objects, but it is not yet clear what the performance will be for complex, self-occluding objects with rapid motions (e. g. humans). Thus, the next step of the research will consist of carefully evaluating the limitations of the strategy, and identifying their cause.

We also plan to use this strategy on a multi-resolution and multi-scale approach, specifically for HCI applications. For example, not every body part requires the same level of modeling. For many applications, the face and hands require more accurate modeling than the torso and legs. Moreover, a single model that incorporates all possible features (fingers, facial expressions, etc.) is not realistic. We would rather use a layered model, based on the same architecture as MBOT, in which each part can be further decomposed into several primitives (see figure 14, taken from [12]). Each level of modeling would require a different resolution, but only in a limited portion of the image. Moreover, the HCI application would dictate the level of modeling necessary for each feature depending on its requirements. Such an adaptive strategy would increase the accuracy of the model and the richness of the interaction without unduly reducing performance.

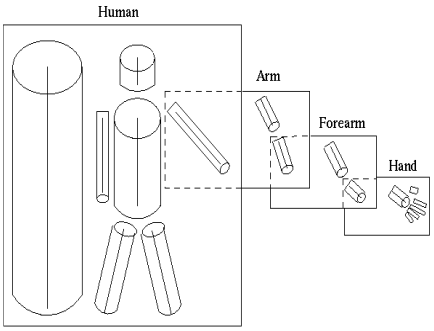

Figure 14: *multi-scale model for multi-resolution processing*

6. Conclusion

We have developed a simple sensor fusion scheme that combines range information with normalized color filtering can be used for fast, reliable feature tracking. We have used this feature tracker to successfully detect a human's head and hands in order to perform a variety of HCI tasks.

We are currently combining this feature tracking strategy with constraints from an primitive-based model in order to achieve fast, reliable and accurate tracking of complex rigid deformable structures. The initial results are promising and demonstrate the validity of the approach. We are now evaluating the limitations of the strategy in terms of model complexity. Our goal is to user this technique to develop a multi-scale, multi-resolution system for real-time human body tracking.

7. Acknowledgement

We would like to thank Roland Siegwart for providing project administration. We would also like to thank the EPFL Institut de Production et Robotique for providing research hardware.

8. References

[1] C. Wren, A. Azarbayejani, T. Darrell, A. Pentland, "Pfinder: Real-Time Tracking of the Human Body", in *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI'97)*, July 1997.

[2] N. Jojic, M. Turk, T. S. Huang, "Tracking Self-Occluding Articulated Objects in Dense Disparity Maps", in *Proceeding of International Conference on Computer Vision (ICCV'99)*, pp. 123-130, Korfu, Greece, September 1999.

[3] D. Beymer, K. Konolige, "Real-Time Tracking of Multiple People Using Stereo and Correlation", submitted to *International Conference on Computer Vision (ICCV'99)*, Korfu, Greece, September 1999.

[4] T. Darrel, G. Gordon, M. Harville, J. Woodfill, "Integrated Person Tracking using Stereo, Color, and Pattern Detection", in *Proceeding of Conference on Computer Vision and Pattern Recognition (CVPR'98)*, pp. 601-609, Santa Barbera, June 1998.

[5] K. Konolige, "Small Vision Systems: Hardware and Implementation", *Eighth International Symposium on Robotics Research*, Hayama, Japan, October 1997.

[6] J. Yang, A. Waibel, "A Real-Time Face Tracker", *Proceedings of the 1996 Workshop on Applications of Computer Vision (WACV'96)*, Sarasota, Florida, USA, December 1996.

[7] M. H. Lin, "Tracking Articulated Objects in real-time range image sequences", International Conference on Computer Vision, September 1999, pages 648-653.

[8] L. Zhao, C. Thorpe, "Recursive Context Reasoning for Human Detection and Parts Identification", IEEE Workshop on Human Modeling, Analysis, and Synthesis, June 2000.

[9] L. Bretzner, "Multi-Scale Feature Tracking and Motion Estimation", Dissertation, Computational Vision and Active Perception Laboratory (CVAP), October 1999.

[10] Fong T., Conti F., Grange S. and Baur C., "Novel Interfaces for Remote Driving: Gesture, Haptic and PDA", *SPIE Telemanipulator and Telepresence Technologies VII*, Boston, MA, November 2000.

[11] R. Tanawongsuwan et al., "Robust Tracking of People by a Mobile Robotic Agent", *GIT-GVU-99-19*, Georgia Tech University, 1999.

[12] D. Marr. *Vision*. W. H. Freeman and Co., 1982.