Real-Time Hand-Tracking as a User Input Device

Robert Y. Wang
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, MA USA
rywang@mit.edu

ABSTRACT

Traditional motion-capture systems have found widespread use in content creation in the entertainment industry. However, their use as real-time user-input devices has been limited by their expense. In this thesis work, we propose an easy-to-use and inexpensive system that facilitates 3-D articulated user-input using the hands, enabling new user-interfaces and applications. Our system optically tracks an ordinary cloth glove that is imprinted with a custom pattern. Observing a custom pattern simplifies the pose estimation problem, allowing us to eschew computationally demanding inference algorithms in favor of a simple and fast nearest-neighbor approach. We investigate optimal design of the cloth pattern as well as possible applications for our system.

ACM Classification: H5.2 [Information interfaces and presentation]: User Interfaces—Input devices and strategies

General terms: Algorithms, Design, Human Factors

Keywords: Hand-tracking, interaction techniques

INTRODUCTION

Recent trends in user-interfaces have seen a rapid increase in consumer devices that capture human motion. These include multi-touch interfaces such as the Microsoft Surface and iPhone as well as motion-sensing devices such as the Wii Remote. In this proposal, we introduce a new motion-capture system that tracks the 3-D position and configuration of a hand. We enable capture of the individual articulation of the fingers in addition to global hand movement. Furthermore, we allow users to move and gesture fully in 3-D instead of along a 2-D surface. We are designing our system with a consumer input device in mind, striving for robust and precise real-time capture with only commodity components.

A commodity 3-D input device would benefit many applications that involve creating or directly manipulating 3-D objects. For instance, controlling animated characters remains a difficult problem partly because any control application must map a low-bandwidth 2-D input device (such as the mouse) to a high-degree-of-freedom 3-D human character. Our system would enable high-degree-of-freedom input directly from the human hand. Modeling applications would also benefit from 3-D input for specifying curves and deformations.

As a consumer input device, our system must be robust and run in real-time. To be useful for most modeling applications, the 3-D input must also be precise. To be accessible to most users of modeling and animation tools, our system must be built from inexpensive, commodity components.

We propose and show preliminary results for a user-input device composed of only a flexible, lightweight cloth glove and one or two commodity webcams. The basis of our technique is that a patterned glove can simplify the pose estimation task. We contribute an image-based nearest-neighbor pose estimation algorithm. We also discuss the optimization of the pattern and applications of hand-tracking.

RELATED WORK

Glove-based and optical motion-capture systems

Glove-based and optical mocap systems have demonstrated precise capture of 3-D input for real-time control. However, these systems are expensive and unwieldy. Glove-based systems may embed dozens of sensors into a glove, which can be expensive and restrictive to movement. Optical mocap systems require obtrusive retroreflective markers or LEDs [12] and many-camera setups. While user-interfaces that use existing glove-based or optical mocap [11, 6, 5] have been prototyped, their deployment has been limited due to the price and setup cost of these systems. Our approach relies on only commodity components and employs a relatively unrestrictive cloth glove.

Bare-hand tracking

Bare-hand tracking continues to be an active area of research. Edge detection and silhouettes are the most common features used to identify the pose of the hand. While these cues are general and robust to different lighting conditions, reasoning from them requires computationally expensive inference algorithms that search the high-dimensional pose space of the hand [16, 15]. Such algorithms are still far from real-time, which precludes their use for control applications.

The bare-hand tracking systems that are real-time are limited in resolution and scope. They may track the approximate position and orientation of the hand and two or three additional degrees of freedom for pose. Successful gesture

UIST'08, October 19–22, 2008, Monterey, CA, USA
ACM 978-1-59593-975-3/08/10.

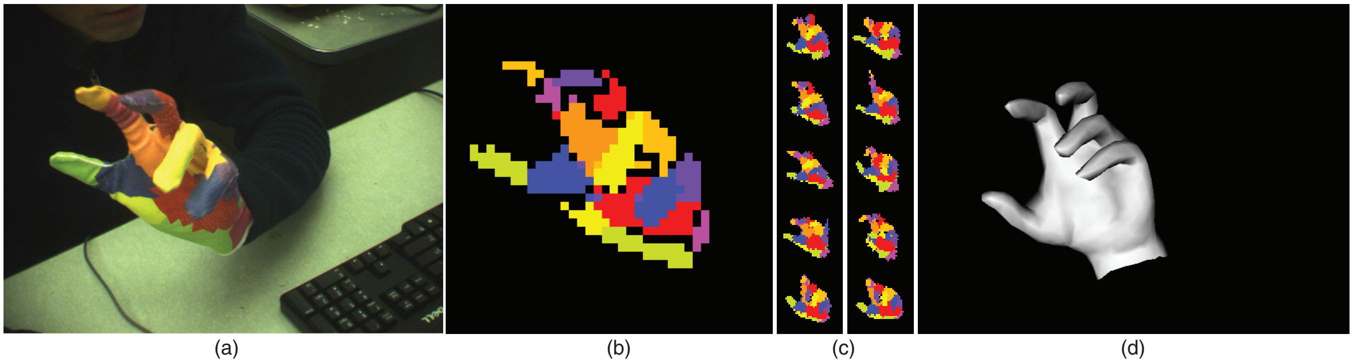

Figure 1: Our pose estimation process. The query image (a) is color quantized, normalized and downsampled (b). The resulting tiny image is used to look up the nearest neighbors in the database (c). The pose corresponding to the nearest database match is retrieved (d).

recognition applications [3, 10] have been demonstrated on these systems. We propose a system that can capture more degrees-of-freedom, enabling direct manipulation tasks and recognition of a richer set of gestures.

Data-driven pose estimation

Our work falls in the class of data-driven pose estimation. Shakhnarovich and colleagues [14] introduced an upper body pose estimation system that searches a database of synthetic, computer-generated poses. Athitsos and colleagues [2, 1] developed fast, approximate nearest neighbor techniques in the context of hand pose estimation. Ren and colleagues [13] built a database of silhouette features for controlling swing dancing. Our system imposes a pattern on the glove designed to simplify the database lookup problem. The distinctive pattern unambiguously gives the pose of the hand, improving retrieval accuracy and speed.

Overall, our design strikes a compromise between wearable motion-capture systems and bare-hand approaches. We require the use of an inexpensive cloth glove, but no sensors are embedded in or outside the glove to restrict movement. The custom pattern on the glove facilitates faster and more accurate pose estimation. The result is a fast, accurate, unrestrictive and inexpensive hand-tracking device.

The remainder of this proposal proceeds as follows. First, we propose a data-driven approach to pose estimation. Second we describe optimizing for the best color pattern on our glove. Finally, we discuss applications enabled or affected by our approach.

POSE ESTIMATION

Many hand-tracking approaches rely on an accurate pose from the previous frame to constrain the search for the current pose. However, these trackers can easily lose track of the hand for good. We focus instead on designing an efficient single-frame pose estimation algorithm. This is possible because the distinctive patterned glove alone is enough to determine the pose of the hand. Traditional tracking techniques can still be used to enhance pose estimation by incorporating temporal smoothness.

In bare-hand pose estimation, two very different poses can map to very similar images. This is a difficult challenge that requires slower and more complex inference algorithms to address. With a gloved hand, very different poses always map to very different images (See Figure 2). This allows us to use a simple image lookup approach. We construct a database of synthetically generated hand poses consisting of different hand configurations and orientations. These rendered images are normalized and downsampled to a tiny size (e.g. 32x32) [17]. Given a query image from the camera, we perform the same normalization and downsampling. The resulting tiny query image is used to look up the nearest-neighbor pose in the database (See Figure 1).

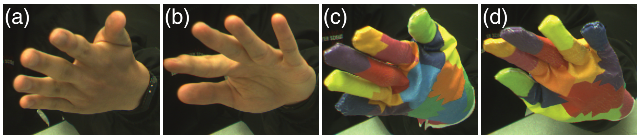

Figure 2: The palm down (a) and palm up (b) poses map to similar images for a bare hand. These same poses map to very different images (c,d) for a gloved hand.

To look up the nearest neighbor in our database, we first need to define a distance metric between two tiny images. We chose a Hausdorff-like distance. For each pixel in one image, we penalize the distance to the closest pixel of the same color in the other image (See Figure 3).

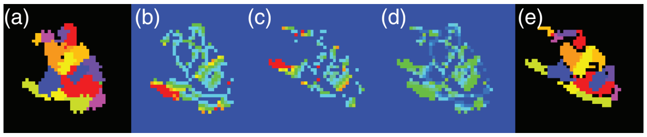

Figure 3: A database image (a) and a query image (e) are compared by computing the divergence from the database to the query (b) and from the query to the database (c). We then take the average of the two divergences to generate a symmetric distance (d).

Given a distance metric, there remain three issues which we address below. First, we describe a method for sampling poses to render for the database. Second, we explore methods of accelerating nearest-neighbor search for real-time use. Finally, we evaluate the relationship between the database size and the accuracy of retrieval.

Database sampling

A database consisting of many redundant hand poses can be inefficient and provide poor retrieval accuracy. We want to sample a set of hand poses that are as different as possible. Our approach is to use low-dispersion sampling of hand poses. The dispersion of a set of samples is defined to be the largest empty sphere that can be packed into the range space (of valid hand poses) after the samples have been chosen. Provided that our nearest neighbor algorithm is sufficiently accurate, the dispersion of the database provides a bound on the maximum pose error that our pose estimation algorithm can make.

We define our distance metric in pose space as the root mean square (RMS) error between vertices of a 3-D skinned hand model. Sampling poses of the hand amounts to drawing hand configurations and orientations that have lowest-dispersion according to this distance metric. We use an incremental low-dispersion sampling algorithm to select our poses. The result is a database constructed so as to minimize the distance from a query image to the database nearest-neighbor.

Fast nearest-neighbor search

Querying a large database for the nearest neighbor is computationally expensive. While our simple Hausdorff-like distance may be robust, it isn't fast enough to be used on millions of images per frame. Instead, we compress each database entry into a short (e.g. 128-bit) binary code for faster retrieval [18, 1]. The codes are designed so that nearby poses are encoded into bit strings with a small Hamming distance. Since these codes are short and the Hamming distance is fast, we can significantly accelerate database search with this approach.

Database coverage evaluation

Given the database sampling and search algorithms described above, we can computationally evaluate the effect of database size on the accuracy of retrieval. For each database size, we measure the average performance of fifty test poses sampled with random hand configurations and orientations. We observe the distance to the nearest neighbor in the database according to the pose distance metric and the image distance metric (See Figure 4). In this case, the pose distance nearest neighbor can be considered the ground truth, while the image distance nearest neighbor is what our algorithm actually retrieves. Both of these quantities become more accurate with database size.

Multi-scale features for pose estimation

We anticipate that our data-driven pose estimation approach can give us pose estimates of within two centimeters of the actual pose. However, for modeling applications with higher accuracy requirements, we may need to perform a few iterations of inverse-kinematics to refine the pose estimate. This can be accomplished by embedding higher-frequency icons and shapes on the pattern. If these features can be robustly located, they can serve as line constraints for inverse-kinematics.

PATTERN DESIGN

The basis of our technique is that a special pattern printed on an ordinary cloth glove can give us a significant advantage in inferring the pose of a hand over bare-hand tracking. This should be true for many patterns, but different patterns can vary in their efficacy. We propose to explore a significant pattern design space from which we can determine an optimal pattern.

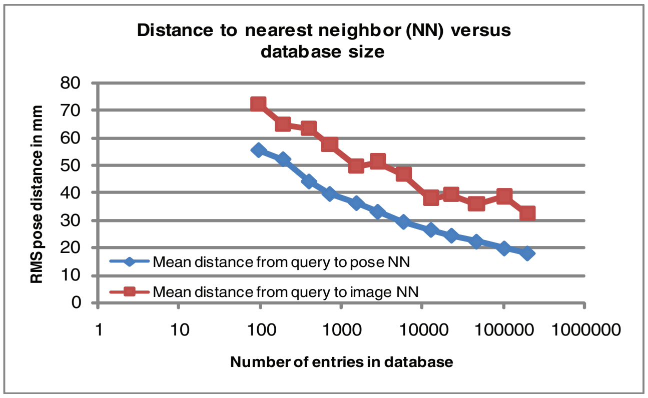

Figure 4: Database coverage evaluation. As the database size increases, average pose distance to nearest neighbors using the pose and image metrics both decrease.

Given our data-driven pose estimation framework, we can evaluate the impact of different cloth pattern designs on the efficiency and accuracy of pose estimation. We begin with a simple pattern composed of twenty, evenly sized blobs of different color. We plan to experiment with the pattern along three dimensions: coloring of blobs, spatial frequency of blobs(See Figure 5), and icons or shapes that can be embedded on the glove .

HAND-TRACKING APPLICATIONS

Our hand-tracking system is intended first and foremost for 3-D direct manipulation tasks. For instance, we would like to enable 3-D modeling in 3-D space by tape drawing of curves or sculpting [8]. We would also like to control 3-D characters by mapping certain points on the hand to points on the character. Laszlo and colleagues [9] have demonstrated direct mappings for animation using low-degree-of-freedom devices. We will explore the applicability of hand-tracking to high-degree-of-freedom animation control.

Many 2-D desktop operations such as filing documents, dragging and dropping elements, and resizing photos also use direct manipulation. These tasks have already benefited from higher degree-of-freedom input via (multi-)touch interfaces [7]. Our system could provide even more expressive extensions with out-of-plane gestures. Distant interaction with large or volumetric displays also rely on hand gestures for selection and "clicking" [19, 6]. We may be able to improve on the error rates of these systems by facilitating richer and more accurate gesture recognition.

The unrestrictive capture of hands enables scientific and artistic applications as well. For instance, we can study how people use physical user-interfaces by capturing the motion of their hands. The hand motions of famous pianists can be catalogued, and their fingerings for particular pieces automatically transcribed. We can improve systems that capture

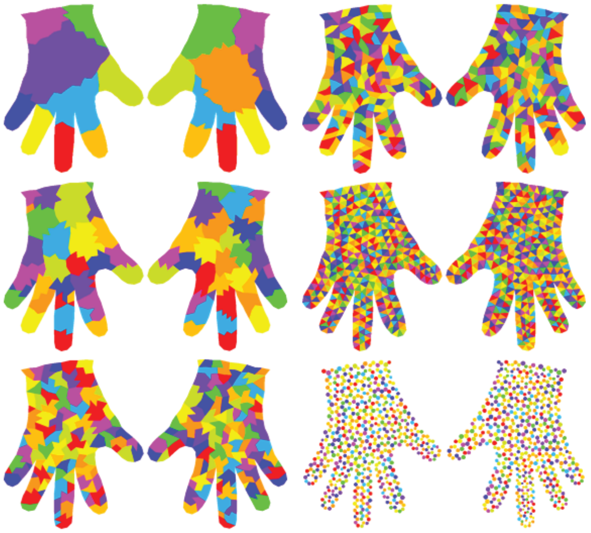

Figure 5: Variations in color assignment and spatial frequency of features on the glove.

or translate American Sign Language, which require precise hand shape, position, and motion sensing [4].

We can envision applications in computer animation and 3-D modeling, new desktop user-interfaces and scientific applications. Many other applications, from virtual surgery to virtual assembly may also take advantage of a hand-tracking system. The goal of this thesis is to deliver a cheap and robust user-input device, providing a stable platform for our colleagues and future users to develop imaginative applications of their own.

REFERENCES

1. V. Athitsos, J. Alon, S. Sclaroff, and G. Kollios. BoostMap: A method for efficient approximate similarity rankings. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2, 2004.

2. V. Athitsos and S. Sclaroff. Estimating 3D hand pose from a cluttered image. *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, 2, 2003.

3. Pushkar Dhawale, Masood Masoodian, and Bill Rogers. Bare-hand 3D gesture input to interactive systems. In Mark Billinghurst, editor, *CHINZ*, pages 25–32. ACM, 2006.

4. L. Ding and A.M. Martinez. Recovering the linguistic components of the manual signs in American Sign Language. *Advanced Video and Signal Based Surveillance, 2007. AVSS 2007. IEEE Conference on*, pages 447–452, 2007.

5. Mira Dontcheva, Gary Yngve, and Zoran Popović. Layered acting for character animation. *ACM Trans. Graph.*, 22(3):409–416, July 2003.

6. Tovi Grossman, Daniel Wigdor, and Ravin Balakrishnan. Multi-finger gestural interaction with 3d volumetric displays. In Steven Feiner and James A. Landay, editors, *UIST '04*, pages 61–70. ACM.

7. Jefferson Y. Han. Low-cost multi-touch sensing through frustrated total internal reflection. In *UIST '05*, pages 115–118, New York, NY, USA, 2005. ACM.

8. Daniel Keefe, Robert Zeleznik, and David Laidlaw. Drawing on air: Input techniques for controlled 3d line illustration. *IEEE Transactions on Visualization and Computer Graphics*, 13(5):1067–1081, 2007.

9. Joseph Laszlo, Michiel van de Panne, and Eugene Fiume. Interactive control for physically-based animation. In *SIGGRAPH*, pages 201–208, 2000.

10. Chan-Su Lee, Sang-Won Ghyme, Chan-Jong Park, and KwangYun Wohn. The control of avatar motion using hand gesture. In *VRST*, pages 59–65, 1998.

11. Jehee Lee, Jinxiang Chai, Paul S. A. Reitsma, Jessica K. Hodgins, and Nancy S. Pollard. Interactive control of avatars animated with human motion data. *ACM Trans. Graph.*, 21(3):491–500, 2002.

12. J. Park and Y.L. Yoon. LED-Glove Based Interactions in Multi-Modal Displays for Teleconferencing. *Artificial Reality and Telexistence–Workshops, 2006. ICAT'06. 16th International Conference on*, pages 395–399, 2006.

13. L. Ren, G. Shakhnarovich, J.K. Hodgins, H. Pfister, and P. Viola. Learning silhouette features for control of human motion. *ACM Trans. Graph.*, 24(4):1303–1331, 2005.

14. G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 750–757, 2003.

15. B. Stenger, A. Thayananthan, P.H.S. Torr, and R. Cipolla. Model-based hand tracking using a hierarchical bayesian filter. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28(9):1372–1384, 2006.

16. Erik B. Sudderth, Michael I. Mandel, T. Freeman, and S. Willsky. Distributed occlusion reasoning for tracking with nonparametric belief propagation. In *NIPS*, 2004.

17. A. Torralba, R. Fergus, and W. T. Freeman. Tiny images. Technical Report MIT-CSAIL-TR-2007-024, Computer Science and Artificial Intelligence Lab, MIT, 2007.

18. A. Torralba, R. Fergus, and Y. Weiss. Small codes and large databases for recognition. *Proceedings of the 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2, 2008.

19. Daniel Vogel and Ravin Balakrishnan. Distant freehand pointing and clicking on very large, high resolution displays. In *UIST '05*, pages 33–42, New York, NY, USA, 2005. ACM.