

Real-time and Robust Grasping Detection

Chih-Fan Chen, Ryan Spicer, Rhys Yahata, Mark Bolas and Evan Suma

USC Institute for Creative Technologies
{cfchen, spicer, ryahata, bolas, suma}@ict.usc.edu



Figure 1. (left) Noise caused by background/sensing (right) The features used for training

Author Keywords

Hand Tracking, 3D user interface

ACM Classification Keywords

H.5.2 User Interfaces: Interaction styles; I.3.7 Three-Dimensional Graphics and Realism: Virtual reality

INTRODUCTION

Depth-based gesture cameras provide a promising and novel way to interface with computers. Nevertheless, this type of interaction remains challenging due to the complexity of finger interactions and the under large viewpoint variations. Existing middleware such as Intel Perceptual Computing SDK (PCSDK) or SoftKinetic IISU can provide abundant hand tracking and gesture information. However, the data is too noisy (Fig. 1, *left*) for consistent and reliable use in our application. In this work, we present a filtering approach that combines several features from PCSDK to achieve more stable hand openness and supports grasping interactions in virtual environments. Support vector machine (SVM), a machine learning method, is used to achieve better accuracy in a single frame, and Markov Random Field (MRF), a probability theory, is used to stabilize and smooth the sequential output. Our experimental results verify the effectiveness and the robustness of our method.

METHOD

We use a Creative Interactive Gesture Camera and the PCSDK [2] to acquire the hand openness, the number of detected fingertips, the average distance from fingertips to the palm, the position of the hand and the normal direction of

the palm (Fig.1, *right*). As the system cannot differentiate between the left and right hand, we use data from a PhaseSpace motion capture system to match data from the PCSDK with the left and right hand. However, adding markers to the hands introduces more noise which causes imperfection of hand segmentation and poor tracking (Fig. 1, *left*).

To increase the accuracy of hand openness, we use the features acquired from PCSDK to train a model by SVM [1]. Although SVM can improve the accuracy of hand openness in single frame, the output label will oscillate and become unusable because it lacks the temporal property. MRF is applied to refine the sequential labels by optimizing the state transition from the confidence of hand openness and previous label. Matching with the hand position from Phasespace, we can use the hand openness of both hands to interact with the virtual environment.

We compared the proposed method with the result from PCSDK and only SVM. The model is trained by using 11918 frames and is tested on 3 video clips with total of 5215 frames. The accuracy is the percentage of frames that have the same state with the ground truth and the transition error is the percentage of the frames with an incorrect state change. In Table 1, although the accuracy of the proposed method will be slightly lower than SVM, it has lower state transition error.

Table 1. Comparison of proposed and other methods

Method	Proposed Method	SVM	Intel SDK
Accuracy	92.02% (4800/5215)	93.90%	73.75 %
Transition error	1.25 % (65/5215)	5.66%	2.07%

CONCLUSION

We presented a method for filtering and fusing data from depth-based gesture camera and motion tracking. Our experiments confirmed the proposed method is more robust and reliable user interface than the compared methods. Its speed for our application and performance make it suitable for many applications such as 3D user interfaces and virtual reality control.

ACKNOWLEDGMENTS

This work is supported by the Office of Naval Research through grant no. W911NF-04-D-0005-0041. The content does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

REFERENCES

1. Chang, C.-C., and Lin, C.-J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology* 2 (2011), 27:1–27:27.
2. Intel Computing SDK. <https://software.intel.com/en-us/vcsourc/tools/perceptual-computing-sdk/home>.