

Conclusion

Improvements and Future

- Gesture recognition algorithm is relatively robust and accurate
- Convolution can be slow, so there is tradeoff between speed and accuracy
- In the future, we will investigate other methods of extracting feature vectors, without performing expensive convolution operations

Accuracy Measurement for Gesture Recognition

| HAND TYPES GESTURE | HAND A ATTEMPT NO. 1 | HAND A ATTEMPT NO. 2 | HAND A UNDER LESS LIGHTING | HAND B WITH HAND A TEMPLATE |
|-------------------------------------|-----------------------------|-----------------------------|-----------------------------------|------------------------------------|
| FORWARD | 90% | 70% | 100% | 74% |
| RIGHT | 96% | 100% | 72% | 88% |
| LEFT | 60% | 92% | 50% | 82% |
| OPEN | 86% | 80% | 72% | 82% |
| CLOSE | 98% | 100% | 100% | 96% |
| AVERAGE | 84.0% | 88.4% | 78.8% | 84.4% |

Frame Rate: 0.4 frames per second

Virtual Reality Explanation

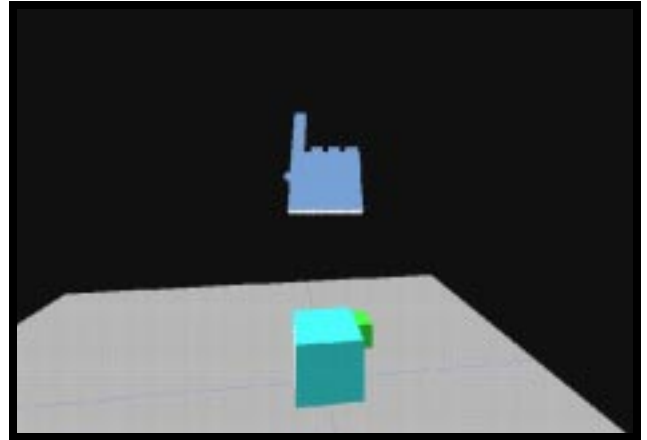
Application

- The user can interact with the virtual environment by hand gestures
- The virtual hand mimics the gesture of the user's hand
- Hand Gesture Commands:
 - Finger pointing up == Moves the virtual hand forward
 - Finger pointing slant == Changes the direction of the virtual hand
 - Closed Hand == Grab an object
 - Open Hand == Release an object
- In the initialization phase, the user supplies the template gestures.
- During the recognition phase, the system matches the sample against the template gestures.

Application 2 of 2



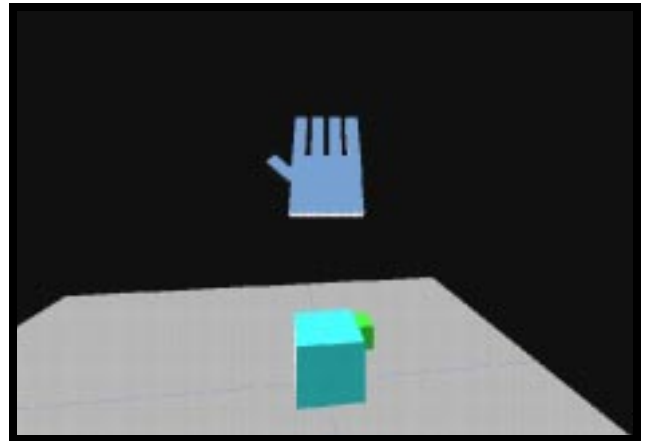
(4) Up



Forward



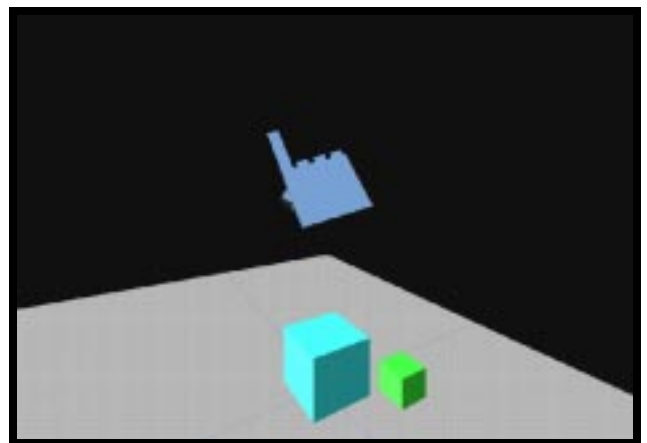
(5) Open Hand



Release



(6) Upper Left

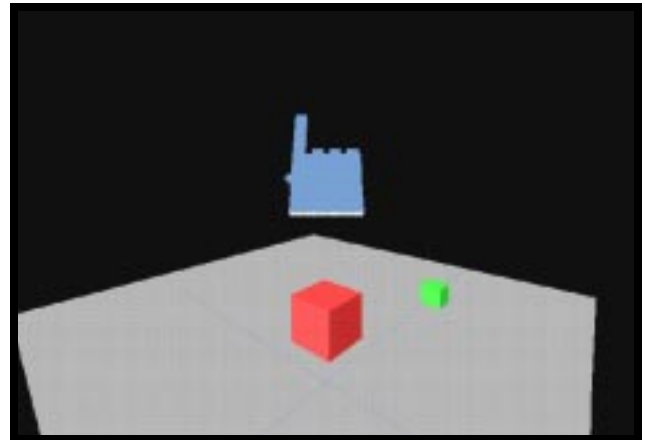


Turn Left

Application 1 of 2



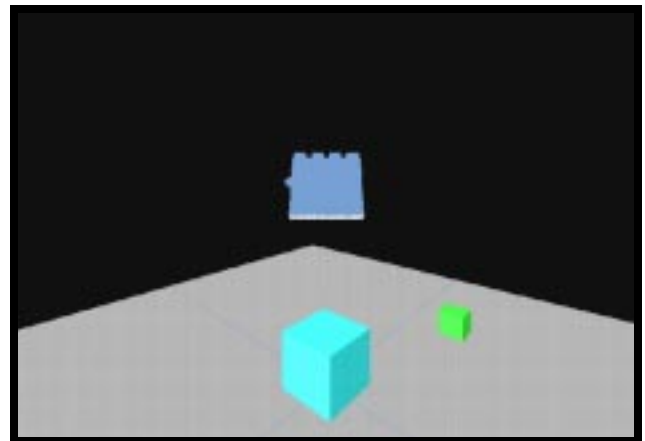
(1) Up



Forward



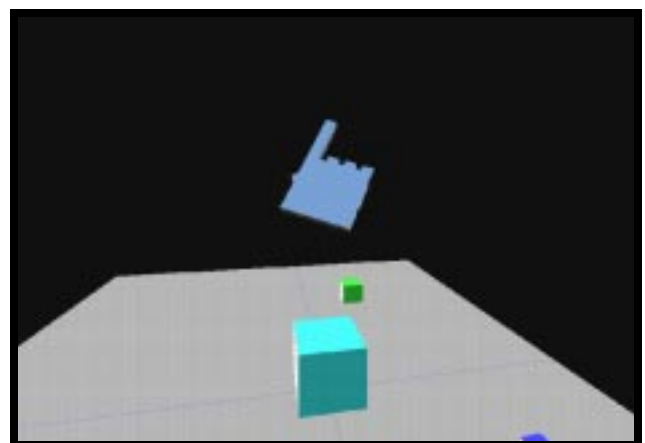
(2) Close Hand



Grab



(3) Upper Right



Turn Right

Composite Imaging Explanation

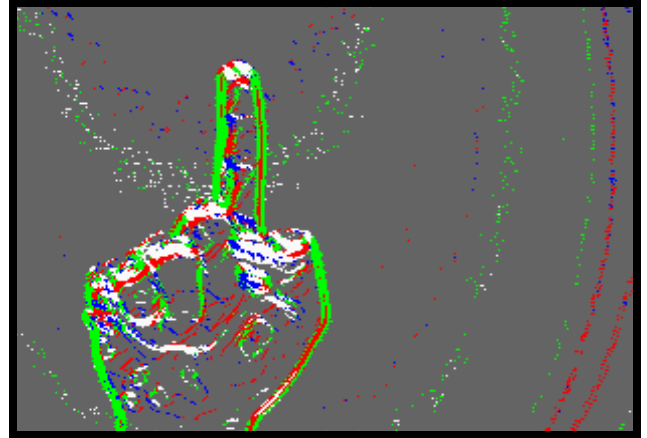
Composite Image Explanation

- We calculated the local orientations at 0, 45, 90 and 135 degrees by convolving the image with appropriate 2-D Gaussian derivative filters.
- We used threshold to eliminate the background noise
- In the figure:
 - Grey == Background
 - White == Local Orientation of 0 Degree
 - Red == Local Orientation of 45 Degrees
 - Green == Local Orientation of 90 Degrees
 - Blue == Local Orientation of 135 Degrees
- The Orientation Histogram is derived by counting the white pixels, red pixels, etc.
- Classification by finding the nearest neighbor with the smallest Euclidean distance to the sample

Composite Image



ORIGINAL IMAGE



COMPOSITE



VERTICAL



45 DEGREES



HORIZONTAL



135 DEGREES

Motivation and Recognition

Motivation

- The user can interact with the virtual environment using hand gestures.
- No Special Hardware Necessary, except for the Camera.
- No Special Hand Markings Necessary

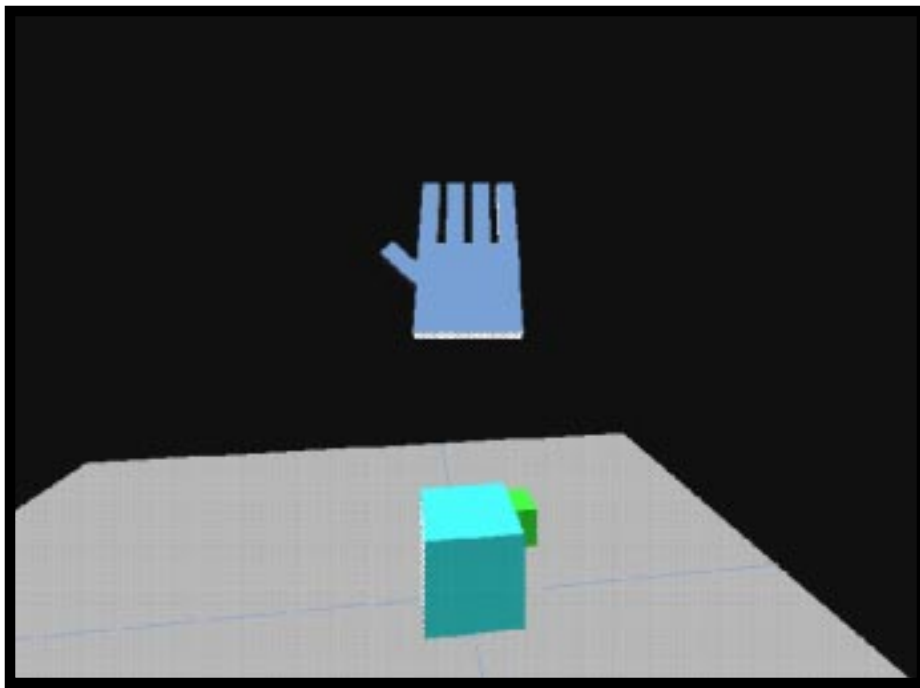
Recognition

We wanted a recognition algorithm that is:

- Relatively simple and fast, which can run in real-time on a workstation
- Robust against changing lighting conditions
- Translation Invariant
- Maintain accuracy, even when different hands are used

We decided to use orientation histogram as the feature vector for gesture classification, since it

Virtual Reality Simulation **using** **Hand Gesture Recognition**



by
Young Cho and Franklin Cho