A Deep Learning Approach for Smart Prosthetics

Erdem Erdemir[1], Matthew Christian[1], Cihan Uyanik[1], Rashad Bailey[2], Erkan Kaplanoglu[3], Kazuhiko Kawamura[4], S. Keith Hargrove[5]

1. Tennessee State University, Department of Computer Science, Nashville, TN USA
2. Tennessee State University, Department of Electrical Engineering, Nashville, TN USA
3. Department of Mechatronic Engineering, Marmara University ·
4. Department of Electrical Engineering and Computer Science, Vanderbilt University
5. College of Engineering, Tennessee State University

Introduction

We present a deep learning classification approach for a smart prosthetic hand that provides an easy and smart grasping capability for the amputee. In this way, amputees will need only minimal visual feedback, which leads to a quick, and easy grasping. The control system was implemented on a powered prosthesis and evaluated using a robot manipulator. Our experiments demonstrate that a deep learning model can correctly classify inertial measurement unit sensor readings while tracing out various grasping trajectories in space. These results show how a deep learning framework can be effectively used for complex IMU motion classification tasks.

Prosthetic prototype

In our work, we chose the Brunel Hand 2.0 from Open Bionics Company which developing low-cost bionic hands shown in Figure 1. The reason why we use this hand as a prototype is lightweight, easy to create using a 3D printer, and is anthropomorphic. The hand design was downloaded and replicated with the MakerBot 3D printer using a polylactide (PLA) polyester. This model has four degrees of articulation in the fingers and is compatible with an Arduino board. Using modified code from the open-source Adafruit Motor Shield, the following commands were programmed: Full Open, Point, Hook, Tip, Spherical, and "Fingers". Using modifications to the sensor collection program and the TensorFlow testing code, a new "prototype" command program was produced. The command program takes the sensor data input by an IMU, creates an image to be fed into the deep learning model, and commands the prototype prosthetic to perform the intended grasp. 100 trials including grasps from each class were performed. Using the best performing network, 80% grasp accuracy was achieved.

Data Collection

We used the Adafruit BNO055 Absolute Orientation Sensor IMU connected via breadboard to a Adafruit FT232H Breakout to convert the GPIO signals to USB. This IMU has 9 degrees of freedom: acceleration vector (Ax/Ay/Az), magnetic field strength vector (Mx/My/Mz), gyroscope or angular velocity vector (Gx/Gy/Gz), absolute orientation (Euler angle Eh/Er/Ep and quaternion angle Qw/Qx/Qy/Qz), and linear acceleration vector (LAx/LAy/LAz). Seven grasp classes were selected: cylindrical, hook, pinch, point, spherical, tip, and tripod. For each class, 100 training samples and 20 test samples were collected. Each sample had on average 79 readings. 840 samples were collected before preprocessing. The subjects in front of a target object with the IMU sensor securely mounted to their right hand using a band. For each collection, the right hand starts resting laterally on the table.

- **Cylindrical:** a large lateral arc
- **Hook:** a lateral arc towards the target
- **Pinch:** a straight reach for the target
- **Point:** Raise the elbow upward & swing the hand forward to point straight ahead.
- **Spherical:** simultaneous pronation & large downward arc
- **Tip:** finger flexors (FDP & FDS) concentrically to move into grasp and isometrically to hold
- **Tripod:** simultaneous pronation & downward arc

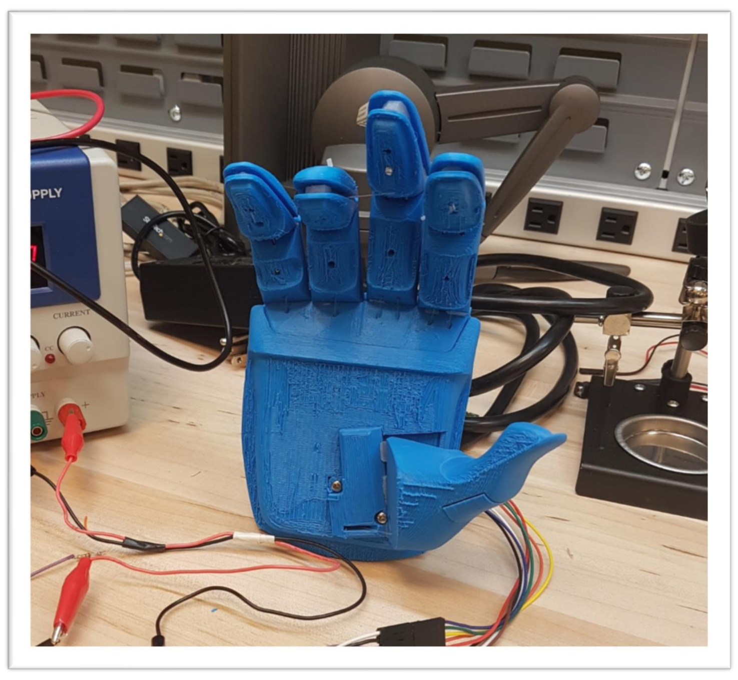

Figure 1. Brunel Hand 2.0 from Open Bionics Company

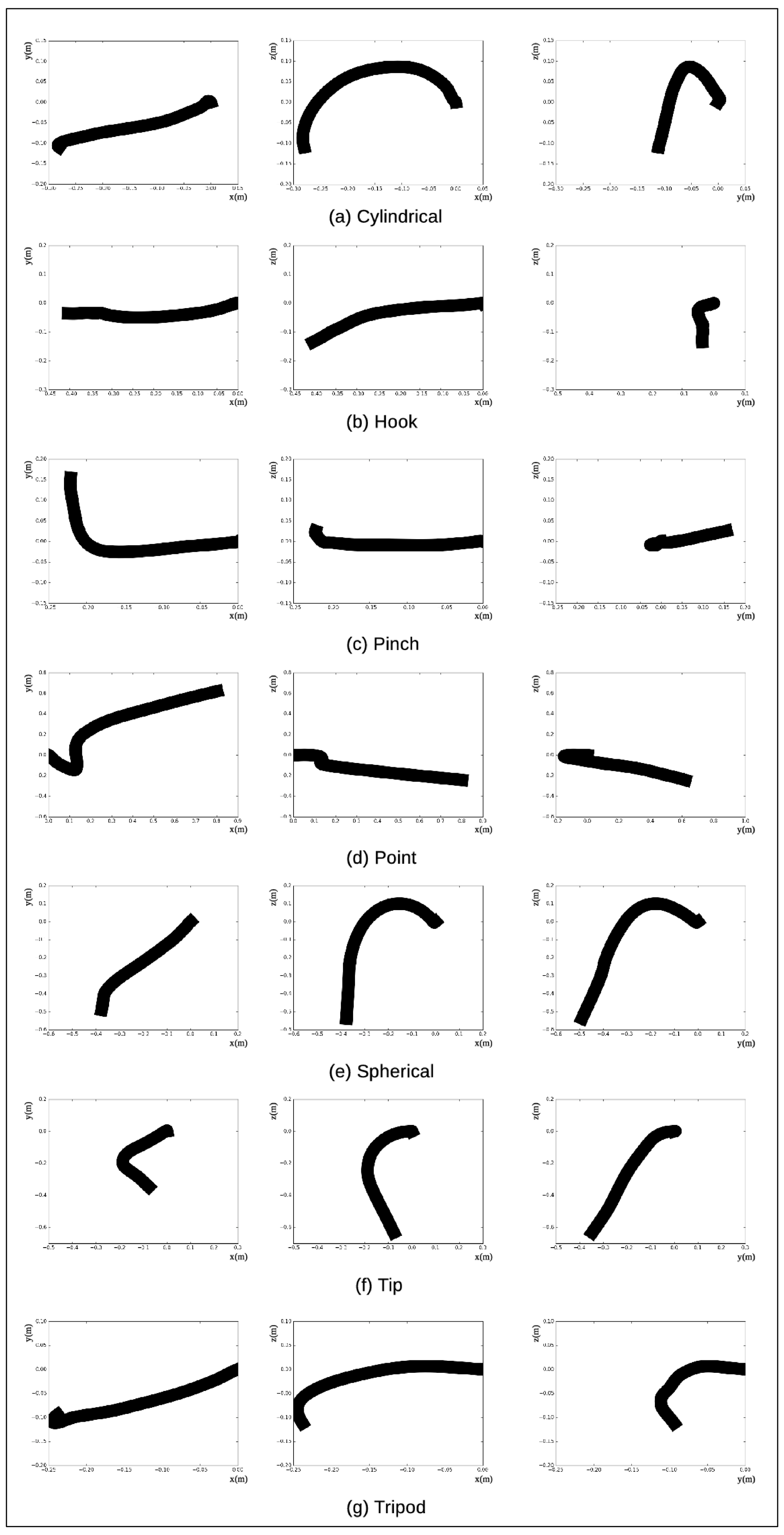

Figure 2. Data Collection from IMU for various grasping techniques

After creating the dataset, the path of motion in 3D space for each sample was preprocessed by integrating the linear acceleration series twice and projecting onto 3 orthogonal planes. This path was rasterized into a 28 by 28 floating point matrix using anti-aliasing interpolation. Floating point value 1.0 maps to a black pixel on the path and 0.0 maps to a white pixel in empty space. The matrix representations along with their labels were written to a TensorFlow Record for further processing.

Deep Learning Models

Three variants of the LeNet-5 neural network were trained.

- **Version 1:** 3 convolutions w/ average pool
- **Version 2:** Version 1 with batch normalization layer
- **Version 3:** Version 2 with L2 kernel regularizer

Figure 3. A Sample Deep Learning Network Frame

These networks were built using the Keras API and run with TensorFlow as the backend. They were trained on the entire training dataset, and each image was processed exactly once. The network training code ran on a Ubuntu 17.10 machine with a 4.00 GHz i7-6700K processor, 32GB of RAM, and a GeForce GTX 980 Ti graphics card with 6GB of memory. TensorFlow was compiled from source with the configuration option to fully utilize the GPU.

Results & Conclusion

The three models were tested on a subset of the collected dataset with promising results. The accuracy of the tests were recorded. As it is seen from our experiment results, deep learning models can be used successfully to classify inertial measurement unit sensor readings while tracing out various grasping trajectories in space. It is easy seen a network with L2 kernel regularizer performs better for the classification task of grasping types.

Table 1. Testing Results of Various Networks

Model	Accuracy (%)
Version 1	75.5
Version 2	93.6
Version 3	95

Future Work

Only accelerometer sensor data has been used to train for classification. Using other data that has been collected can increase the accuracy of the training. Also new contemporary deep neural networks can be used to improve the performance.

BARlab
Biomechatronics...
Agriculture...
Robotics...

https://bar.tnstate.edu

Tennessee State University