

# *Wand free spell caster*

## *Hand gestures control API for electronic devices*

### **1. Context and motivation**

In today's world, we are constantly surrounded by a huge diversity of smart devices, all which are designed to help us in our daily routines and to make our lives better. However, I believe that they all suffer from a great disadvantage as each device comes with a different way of controlling and communicating with it, which results in users who needs to learn and remember a lot of different commands and buttons functionality.

In order to solve this problem, a standard method of human-device interaction should be established and implemented for every smart device. And, in my opinion, the best approach would be to use hand gestures.

Hand gestures are a natural communication method, used by people from an early age due to the vast quantity of information that can be exchanged and transmitted through this approach. They are easily performed by all humans, and should be easily recognized by smart devices that have a camera and the necessary software.

### **2. Proposed solution**

As I have already mentioned above, the purpose of our next project would be to develop the software that would allow smart devices with a camera to respond to hand gestures. The solution I would like to propose you, consists of researching, designing and developing a library and an API that would be integrated with smart devices and would allow them to detect and recognize hand gestures and to act accordingly to these. The software that we are going to develop should have three major components: a component for detecting the hand gestures, a component for recognizing those gestures, and a component that would allow the smart device software to respond based on those gestures. This project will have a 6 months milestone, when we are going to present a functional implementation that could be deployed on a smart TV, a tablet and light controlling system. The algorithm should be able to recognize and respond in real time to a number of approximate 15-20 hand gestures in a variety of lighting conditions, textures and resolutions.

### **3. Challenges**

In order to achieve the desired results and have a working prototype in 6 month from now, I have already identified a number of challenges and aspects that must be considered and solved when developing the library and API.

First of all, we need to consider the machine learning algorithm that we are going to use for hand detection and recognition. Because these tasks are Computer Vision

tasks, I recommend to use a Convolutional Neural Network architecture. However, such an approach requires a lot of data to learn from, and it usually becomes slower as the neural network's complexity increases. As a result, an important step of our development process would consist of collecting a significant dataset for training the algorithm. There will also be necessary to experiment with multiple CNN's architectures in order to optimize the speed - performance ratio. Another important problem comes from the fact that the hand is an unstable object. The change from one gesture to another would cause a series of transient, intermediate gestures, to whom the system should not make any response. Moreover, the hand is a highly deformable object, that can easily suffers important changes in appearance, shape and color. As a result, our system must be robust to all of these transformations and must be reliable, and tolerant to false positives.

#### **4. Roadmap for the next six month**

- In the next two weeks:
  - i. Research the SoA approaches for hand detection and gesture recognition
  - ii. Select the 15-20 gestures that we must be able to recognize
  - iii. Determine the main constraints that should be taken into account so that we will have a working prototype that will fulfill the requirements described in section 2
  - iv. Decide what technologies we are going to use in term of programming language, frameworks etc
- In the next three months:
  - i. Collect a sufficiently large dataset ( ~ 300k images) for training the Machine Learning models.
  - ii. Develop the software architecture
  - iii. Train at least 5 different models and compare their accuracy, robustness, speed and memory constraints
  - iv. Start implementing the integration module for a smart TV, a tablet and a light controlling system
- In the next five months:
  - i. Start working on ML models optimization; try reducing the parameters and improving the speed, while retaining the accuracy ( a maximum of 3% loss is acceptable)
  - ii. Work on deploying the full pipeline on one of the smart devices used for the Demo Day
  - iii. Test the implementation ( I want the software to be deployed and tested in real life conditions)
- Six month from now - Demo Day