

Hand gestures control development plan

Analyzing, designing and implementing software solutions that integrate control of several environments through the users' hand gesture is an elaborate process that requires thorough research, foresight and planning. This document summarizes a few of the challenges the team might face while developing such applications as well as a comprehensive list of steps towards delivering a first prototype.

The first step to be taken revolves solely around considering the start-up's mission and objectives, as well as the area covered by the existing AI and Machine Learning products functionality, in order to estimate a baseline user workflow and anticipate a rough list of behaviours and gestures. To justify the importance of this kind of investigation one should imagine that an interface for hand motion control implemented in projects of companies that mainly focus on game development, for example, would have to handle a significantly larger and more different set of movements than if it were designed, say, for user accessibility purposes, like translating sign language to ASCII text. Another aspect that should be covered in the research phase of the project, and can be even **one of the most impactful challenges** to overcome consists of the way the algorithm and storage strategy makes use of the hardware resources. Therefore, an evaluation of possible platforms along with their specifications and limitations is required for the successful conception of software with efficient resource management.

Secondly, data acquisition is another thing to look into, and that itself rises **another challenge**, since the supported platforms have not been decided, and it would be ideal to cover as many of them as possible. The best way to facilitate this is by splitting this functionality into smaller independent modules, this way covering **one of the implementation problems** that regards the modularity, scalability and maintainability of the code. Consequently, one module would define the data structures and model format of the information received as input from the other module. The latter will strictly deal with matters of reading the signals coming from the hardware equipment, parsing it and passing it along in the correct format to the defined information model. This ensures the possibility of providing support for multiple platforms without having to modify the way the information is represented in our program.

The third step would be defining the motion library and deciding languages and technologies to be used in its implementation. At this stage the information gathered previously should be taken as input by a new module that makes sense of it, mapping the information in the model to actual physical motion. This takes us to the next point of this stage, which is the need for a new model, a new data structure that centralizes the information from the previous model into a collection of individual movements with a speed, a set of coordinates, a trajectory and an attribute referring to its proximity of the RGB camera, for example. The library transforms the variation of the model properties described in step two into a more intuitive set of information, easier to give significance to and to use it to trigger a reaction in the system's

behaviour. An **implementation problem** that might arise and is worth mentioning is that if any third party library is needed, it needs to have continuous, effective and updated support. Moreover, **a third and very important challenge** that also regards **implementation** lies in the definition of this new data structure itself and being able to identify and differentiate one movement from another with enough precision.

Step four brings the prototype's development closer to an end and provides a wrapper around the data acquisition module and the motion mapping one, to be used as a whole when implemented in another software's environment. Each gesture defined in the previous model should be assigned a meaning at this stage and should output a standardized signal that triggers a reaction in the system that reads it. The same hand gestures can mean different things in different contexts (i.e. bringing the index finger and thumb together could depict the "zoom-out" action in the context of browsing through a web page, or the "lower-volume" action when the library is included in a motion controlled home-cinema system), so their response should be defined when implemented within the software that needs the feature. Furthermore, providing a standard output that respects a specific format smoothens the compatibility with other platforms.

Fifth and final step consists of building an error logging tool designed to detect improper or unclear input from the user and alert them with an appropriate signal depending on the platform it is implemented and the type of error that is being made.

Eventually, testing and documenting the tool are very important parts of the development process for both the maintenance and expansion of the tool. These two tasks, however, need not be mentioned in a separate step of the development plan as each part of the project should be tested and documented as soon as it is completed.