2.5 Summary

This is the final part, where analysis, accuracy considerations and further development options will be discussed.

2.5.1 Results

As a result, the following features were developed:

1. Coordinate system – now it is possible to manipulate the robot arm with the help of a single coordinate. For efficiency, a coordinate system has to be defined and further the robot arm will follow the reference point.

2. Machine learning platform – the robot can now be provided with AI capabilities. It is possible to add functionality by means of Agent scripts and training with the help of external Python API. It is also possible to improve/reduce the accuracy with the help of hyperparameters tuning.

These features, together with the research on the implementation of ML in a virtual environment will serve as a platform for further development of more complex systems, with broader functionality and improved precision.

2.5.2 Accuracy of the solution

At the current stage, this development platform is not optimized for usage with real robot arm due to the fact, that the accuracy is poor. This is due to difficulties that arose during the development stage of both practical coordinate and decision system implementations.

Coordinate implementation accuracy. During the development of the machine learning system, additional difficulties and questions related to the coordinate system appeared, such as whether the robot arm can grab the object or difficulty with following the reference point with precision during the object moving. These features are important, if the actual working prototype is to be developed, because even though in Unity it is possible to either make a child/parent relation or fine-tune some parameters for BioIK script, in real life, such constraints may cause unexpected behavior.

As a solution, there is a possibility to create a different machine learning algorithm for moving to a specified coordinate: it would move the reference point incrementally checking how much the joint angle values are changing. As a result, it would be expected to have minimum time with the minimum path to reach a coordinate. However, this method may have limitations in terms of performance, as it would need a machine capable of powerful computing. For second problem it could be implemented so that a system is able to grab the cube regardless of its orientation based on physical properties only (thus limiting the additional parent/child features and still managing to translate the object in space), which will enable the robot to adjust the gripper according to the position of the cube.

Machine learning implementation accuracy. Here the accuracy is relatively poor, however, provided more time given and better expertise regarding machine learning algorithms and reinforcement learning specifically, this would be possible to reach the desired accuracy. This part does need accurate decisions, as the robots are expected to be precise in any sphere where they are used. Problems that occurred were also dependent on processing power and if the learning process could be sped up significantly, the results would be obtained faster and therefore the accuracy would be higher (assuming the parameter tuning was taking place).

2.5.3 Further development plans

For further development there are several challenges that may be solved:

1. The accuracy problem – this would still need to be solved
2. Implementation of knowledge translation to the real robot – this will be explained further
3. Additional tasks – increase the functionality of the robotic system of autonomous decisions
4. Objective development strategy – a specific set of instruction on how to implement any new complex task

The accuracy problem was thoroughly discussed in the previous item, but there are three new challenges, which should be described better.

Implementation of knowledge translation to the real robot. The nature of this thesis is to implement functionality for a digital twin, and this fact assumes that the information is transferrable to a real robot. Even though there was no possibility to test the behavior of the current solution on the real robot, there is a need to do so in the future and develop the connector between Unity VR interface and real robot interface.

Objective development strategy. Currently, the development process roughly depends on a single developer. Implementing new features with machine learning algorithms should be more decentralized with the possibility to have a team of developers working on developing smart systems for a digital twin, each developer working on a specific task. Therefore, it is needed to define an algorithm for developing the functionality and define a system, which would make this process modular, even if there are components that depend one on another.

Additional tasks. A single cube color sorting task for one cube would not be sufficient in the industrial scale. For a product to be competitive and useful, there is a need in additional features, whether they are the complexity of present tasks (introducing more cubes, more colors, more shapes, etc.) or new, not developed before tasks (advanced positioning, building towers with objects, processing of an object, etc.). Additional tasks that would be developed may bring more value to the product by decreasing the costs of workforce and maintenance.

2.5.4 Economic point of view

Developing such a system may not be easily conducted on a regular manufacturing site. A smart digital twin is a complex system that must be carefully replicated and programmed. This item will investigate the potential cost of the development of such a system.

According to the worst-case scenario, the following table may represent a potential setup and maintenance cost for such a system as in Table 2.5.4.1.

Table 2.5.4.1. Bill of elements needed for setting up the system.

Type of product:	**Quantity:**	**Price per unit (estimated in Euro):**
Workforce (per one developer/month)	5	2,109
NVIDIA TITAN RTX	1	2,302
VR Equipment	2	366
Office space (per square meter)	35	210
TOTAL:		**20,929 EUR**

As per the table, it is possible to see, that the setup price (which normally would not take more than one month) is worth 20,929 EUR according to the information which was found from statistics and official sites of products. This sum is approximate, as the availability, delivery price, and economic conditions may vary, but generally, the price was estimated for Estonia.

In order to maintain the system monthly (if such need is present), the developers and office space for them should be provided, which would total in 17,895 EUR. This sum is calculated without considering the growth of the department and is paid monthly.

The explanation for the products is given in Table 2.5.4.2.

Table 2.5.4.2. Explanations for needed elements.

Workforce (5 developers)	It is needed for setting up and programming of the system, this is the minimum requirement for the system to work. It is expected that 5 developers would be enough, if there are: 1 CAD-developer, which would transfer real environment to virtual environment; 1 ML-specialist, who would develop machine learning systems; 1 UX developer – for creating an application which operators can use; 1 backend developer, who would combine ML module, UI module and connection with machinery (actually making a digital twin) and finally, a project manager, who would transfer business needs to more technical terms. The salary was taken as average per year of 2018 [11].
NVIDIA TITAN RTX	This is a GPU that was designed specifically to work with AI [12]. Considering that there will be a lot of different machine learning, it would be important to provide performance and therefore speed, not to slow down the

	development process and start saving costs faster.
VR Equipment	Working with VR is essential here, because ideally machines will do everything by themselves, and an operator will only handle tasks in the virtual environment. Equipment (Oculus Rift S [13]) would be needed to test and teach operators for further interaction with systems. Later more equipment can be bought if there are more operators needed.
Office space	Ideally, the recommendations are, that for each employee there should be $5m^2$ of space and the price was taking according to the statistics [14] of price per m^2 of office space.

To be able to calculate profits, which the company could make, it would be necessary to calculate the costs that could be saved with this system, however, as this thesis looks into the approach to the problem and not a specific implementation, profits should be calculated on a case basis, as there are many different spheres where this approach may be used.

2.5.5 Potential usage

This technology can theoretically deal with many different problems ranging from lack of personnel (for example if there are several machines with simple repetitive tasks it would be possible to set them up for independent actions) to development of new equipment, which would take smartness of a system as a basis.

There are several main industry sectors which could benefit from using digital twin and therefore from using machine learning in combination with it:

- Manufacturing industry – this was already discussed in the context of this project, as the testing was done on a robot used in the manufacturing sector. One of the best use cases for machine learning implementation, as machines on manufacturing sites usually need operators, and with machine learning capabilities these requirements may be reduced to a minimum.

- The Healthcare industry – this sector also is being used with combination with digital twin concept and providing ML functionality may substantially help or speed up processes in this sector. For example, it would be possible to create equipment with a scanner, that would scan the body by itself, therefore generating a full model of the body for further suggestions to the real doctor. This system would make it easier to analyse a person's body and consequences that may arise after doctors' actions.

- The entertainment industry – this is one of the biggest sectors in today's world and it can use plenty of ML capabilities in conjunction with the digital twin as well. In fact, ML-Agents was designed for game developers, so it would be even easier to implement functionality for this sector, therefore creating new content that will be profitable.

However, these are sectors, which are generally considered use cases for digital twin technology and therefore may adopt ML potential regarding digital twin development. There might be more use cases in other disciplines, but more research is needed for this matter.