המחלקה להנדסת תוכנה

שם הפרויקט: מערכת לזיהוי מנח כף יד ואצבעות בזמן אמת

Project Name: Handy

הגדרת הפרויקט – Statement Of Work

שם הסטודנט: ניר בן דור

מספר תעודת זהות: 204588388

שם הסטודנט: דניאל קורניס

מספר תעודת זהות: 203439997

יועץ מומחה: אדם אשר עוזר לסטודנט בתחום אשר דורש מומחיות מיוחדת, עזרה (של לפחות 5 שעות, אם אין כזה יש למחוק את השורה)

תאריך ההגשה: _____

שם המנחה: אהוד דיין

חתימת המנחה:

חובה להחתים את המנחה לפני שהמסמך מוגש. החתימה מציינת) שהתוצר אושר על ידו. יש לסרוק את הדף החתום ולהוסיף כעמוד הראשון של התוצר. ניתן לחילופין לצרף עמוד שני למסמך עותק של מייל מהמנחה לפיו הוא מאשר אותו)20343997

ED Ehud Dayan <ehud.dayan@gmail.com>
Sun 7/12/2020 4:27 PM
To: Daniel Kornis

שלום דניאל וניר
ה- SOW מאושר להגשה

--

Sonarion LTD , CEO
www.sonarion.com
udi@sonarion.com
Cellular phone: +972-544-860435
Office: +972-544-860435
Fax: +972-151-544860435
Skype : Sonarion
Yehuda burla 26 / 12
Jerusalem, Israel

Reply | Forward

Table Of Contents

List Of Figures

List Of Diagrams

List Of Tables

4. Introduction

4.1. Problem Definition

Human-Computer interface has a large influence on productivity and ease of use of today's tools and computers, and in recent years there were several attempts to expand new technologies to fields that require delicate and precise handling, such as Robotics for surgeries, Virtual Reality Gear, heavy machinery and others.

For this reason, there is a need for a better, more precise and intuitive interaction between humans and computers. Equipment today is often clumsy and accompanied by surrounding hardware which makes it more expensive than it could be.

Our Project will be an intuitive and relatively cheap alternative that allows precise and delicate movement in real time. It harnesses Machine-learning architecture to extract hand and joint poses from a live input stream of RGB-D photos.

In our project we will introduce a deep learning network in addition to data extraction from live input stream.
We would also implement a preprocessing stage prior to the insertion of the data into the trained model.

4.2. Purpose And Objectives

The purpose of the project is to develop a new and more precise way to integrate the human hand motions and the virtual / hardware implementations of the command.
The main objective is to produce a functioning deep learning system that is capable of learning from a pre established dataset and predict the positions of the hand and all of its joints in order to transfer them to a seperate system.

5. Project Goals and Objectives

5.1. Objectives

Construct a robust system that utilizes machine learning inorder to extract hand and finger poses in 3D space from RGB-D images. The system will allow the human hand to function as an input for many different operations, whether interacting with a virtual environment or translating human gestures to mechanical joints.

We would also intend on building a virtual hand by using the Unity Game Engine in order to test our project and simulate the model's predictions in a virtual environment.

5.2. Functional and Quantitative Goals

Project Functional Goals:

- Locating the movements of a hand that operates a joystick from RGB-D images
- Overcoming the technical difficulties that arise from situations where the targeted hand is obscured by other irrelevent objects.

Project Quantitative Goals:

- Processing the Images at a rate of 10-20 HZ.

5.3. Measurements:

Project Measurements :

- Image Processing Rate (in Hertz)
- Precision of the joints extracted compared to the ground truth and the existing competition (in Millimetres).

6. Literature Review

6.1. Introduction

Literature review provides a broad view of the problem at hand - integration between man and machine in the field of hand recognition. Some of the papers propose solutions to similar issues and provide an outlook on how different approaches perform, with a wide perspective on the gap between the performance of current technology trends and the goal at hand.

We will also discuss our approach of combining the separate fields of machine learning and computer vision in order to solve the problems that arise from said integration.

6.2. Scope of the Problem

Papers:

- **Efficient Hand Pose Estimation from a Single Depth Image:**
  This paper tackles the practical problem of hand pose estimation from a single noisy depth image with a dedicated three step pipeline, the approach is able to work with Kinect-type noisy depth images, and reliably produces pose esti-mations of general motions efficiently

- **Rule Of Thumb: Deep derotation for improved fingertip detection:**
  This paper tackles the problem of per-frame fingertip detection in depth images. The method reduces the complexity of learning in the space of articulated poses. The approach also describes a pipeline for high accuracy magnetic annotation and labeling of objects imaged by a depth camera.

- **First-Person Hand Action Benchmark with RGB-D Videos and 3D Hand Pose Annotations:**
  This work presents an experimental evaluation of RGB-D and pose-based action recognition by 18 baselines/state-of-the-art approaches. The impact of

using appearance features, poses, and their combinations are measured, and the different training/testing protocols are evaluated. Finally, we assess how ready the 3D hand pose estimation field is when hands are severely occluded by ob-jects in egocentric views and its influence on action recog-nition. From the results, we see clear benefits of using hand pose as a cue for action recognition compared to other data modalities. Our dataset and experiments can be of interest to communities of 3D hand pose estimation, 6D object pose and robotics as well as action recognition.

6.4. Initial Market Survey

The SIMI german company and others use multiple cameras around a set frame, a long tiring process which requires operator involvement and is not in real time. Other companies, such as UltraLeap use stereoscopic cameras and private, corporate algorithms that are not accessible to the public.

6.5. Existing Products / Competitors

The main competitors are:

- **Leap Motion** :
  The Leap Motion **controller** is a small USB peripheral **device** which is designed to be placed on a physical desktop, facing upward. It can also be mounted onto a virtual reality headset.
  Using two monochromatic **IR cameras** and three infrared LEDs, the device observes a roughly hemispherical area, to a distance of about 1 meter.
  The LEDs generate pattern-less IR light and the cameras generate almost 200 frames per second of reflected data.

  The data is then sent through a USB cable to the host computer, where it is analyzed by the Leap Motion software. The software uses **calculations** in order to compare the 2d frames generated by the two cameras and synthesize from it the 3D position data.

- **Oculus Quest** :
  The Oculus Quest is a wireless virtual reality **headset**. It doesn't need to be tethered to a PC and doesn't require a phone.
  It features six degrees of freedom (6DoF), meaning it can track up, down, left, right, forward, and backward movements. It doesn't require any external sensors, Instead, it has **sensors** built into the headset. It also supports two updated **Touch controllers**.
  A **camera** in each corner of the headset (total of four) **track** space and motion controllers from the inside out.

  The technology behind it makes use of wide angle cameras and **hardware** based image analysis to identify landmarks within a room and then **determine** the location and orientation of the headset based on how those landmarks appear. It uses a similar method to track the controllers, though instead of orienting them relative to the landmarks it orients them relative to the headset.

- **Wrnch** :
  wrench is a computer vision / deep learning software engineering company based in Montréal, Canada, a world-renowned hub for AI and visual computing. The wrnchAI platform enables software developers to quickly and easily give their applications the ability to see and understand human motion, shape, and intent.

  The technology behind wrnchAI is based on human motion capture and activity recognition.
  Human motion capture digitizes human motion, allowing machines to track or reconstruct human behavior. The main advantage of human motion capture is that large amounts of data can be processed within a few milliseconds. This enables applications to perform in real-time, such as movement analysis for sports and automation involving human-machines interactions.

  Motion capture is performed via **joint skeletal tracking**, which tracks humans in a video by creating a virtual skeleton overlay. The skeleton consists of several skeletal joints and segments, representing the body parts and limbs. The number of skeletal joints can vary according to the pixel resolution, which can vary depending on how far an individual is from the camera. The timeline of skeleton point and segment coordinates forms the digitized human motion data from which movement paths and trajectories can be estimated.

Activity recognition involves tracking an individual over time as they perform a series of actions. The machine learning model compares the ongoing action to the set of actions that it was trained on, allowing it to not only recognize the actions but also assess movement deviations by comparing against the average trajectory.

- **Valve Index** :
  The headset uses LCD panels for each eye - the panels are full RGB and can operate at refresh rates of 80 Hz, 90 Hz, 120 Hz.

  Central to the Lighthouse technology are the Base Stations. These Base Stations are small rectangular objects placed in the **tracking area.** They serve as reference points for any positionally tracked devices such as the HMDs and controllers. Base Stations perform this function by constantly flooding the room with a non-visible light. The **receptors** on the **tracked devices** would intercept the light and figure out where they are in relation to the Base Stations. Multiple Base Stations (2 for SteamVR) allow the tracked devices to figure out where they are in the 3D space.

  Each Base Station contains an IR beacon called Sync Blinker and 2 laser emitters that spin rapidly. 60 times per second, the Sync Blinker would emit a synchronization pulse and 1 of the 2 spinning lasers would sweep a beam across the room. The receptors, HMDs and controllers, are covered with photo **sensors** that recognize the synchronization pulse and the laser beams. When it **detects** a synchronization pulse, the receptor starts to count til one of its photosensors is hit by the laser beam. Lighthouse **calculates** When the photosensor is hit by the laser and Where that photo sensor is located to find the exact position of the receptor in relation to the Base Station. When there are 2 Base Stations, the position and the orientation of the receptors in the 3D space of the room is established.

  Base Stations are **vulnerable to occlusion**. They require line of sight to the tracked objects. Base Stations are designed to be scalable. 2 Base Stations are placed in opposite sides of the room to minimize this problem. More Base Station can be placed to increase the tracking range.

6.7. Comparison

Category	Leap Motion	Oculus Quest	Valve index	Wrnch	**Handy**
Price					
Spacial stability					
Surrounding hardware					
Ease of use					
Mobility					

Good	**Mediocre**	**Bad**

6.8. Advantages of our Product

From the initial market survey that we've conducted we found out that all of our competitors were using sensors and wearable gear (headset, controllers, gauntlet ,etc.) In order to track the hand's position and movement.

Our product will nullify the need to use such equipment which will further improve the user's experience and immersion into the virtual environment.
The lack of wearable devices can also contribute to a more natural hand movements and more precise actions that are not available for the user in the present - like grabbing virtual objects, turning keys and tightening a grip around ingame tools and devices.

Moreover, The fact that the client no longer needs to rely on wearable/surrounding gear will reduce the total price of VR games and will attract new potential customers who couldn't afford it before.

7. System Requirements

7.1. Functional Requirements

The software will accept input of RGB-D Images that will be processed in a neural network and output three dimensional coordinates for hand and finger positions.
This project will require a small-medium sized dataset of RGB-D images with labels for the positions of the hand and joints.

Options available to the user would possibly include:

- Different types of preprocessing that may improve detection against different backgrounds.
- Performance options to balance speed of predictions agianst the accuracy of predicted poses.

7.2. Non- Functional Requirements

- The predictions of the trained model will be visualized using a Unity based virtual environment which won't require any prior knowledge in a specific field.

- The data will be entered in a consistent format and resolution

- User friendly, Simple and intuitive API.

8. Block Diagram

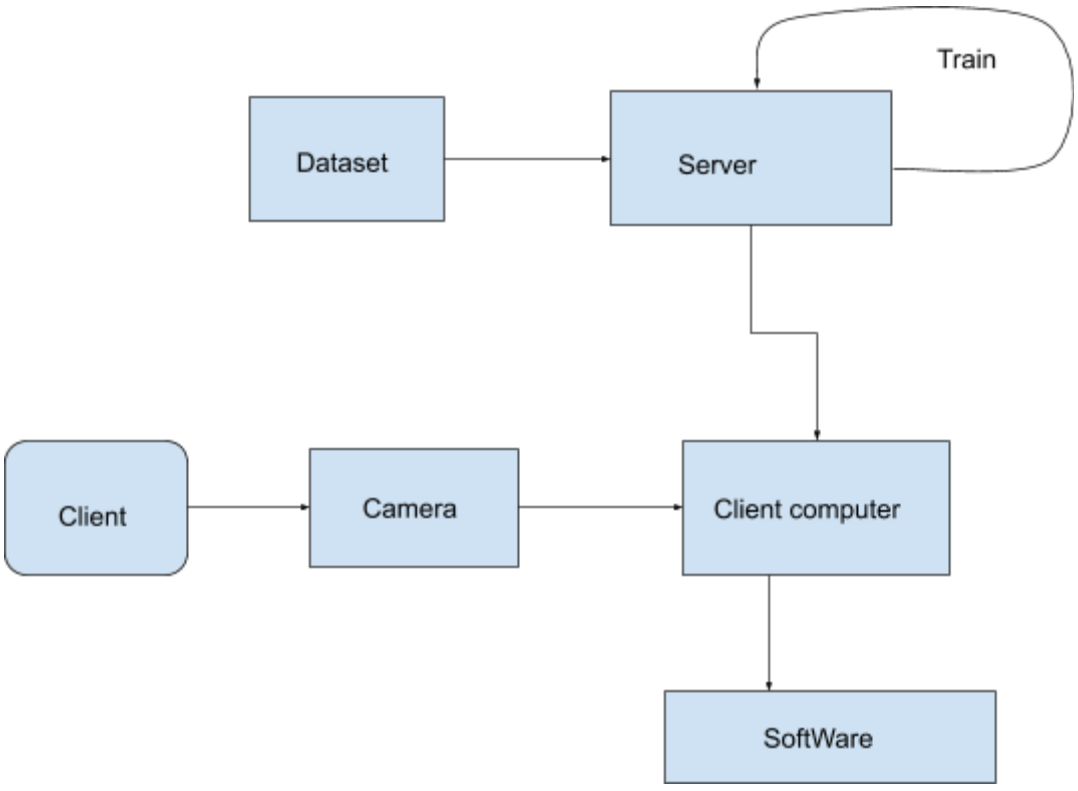
Architecture Diagram:

- RGB-D Camera: A camera with three color channels and a depth channel to record images.

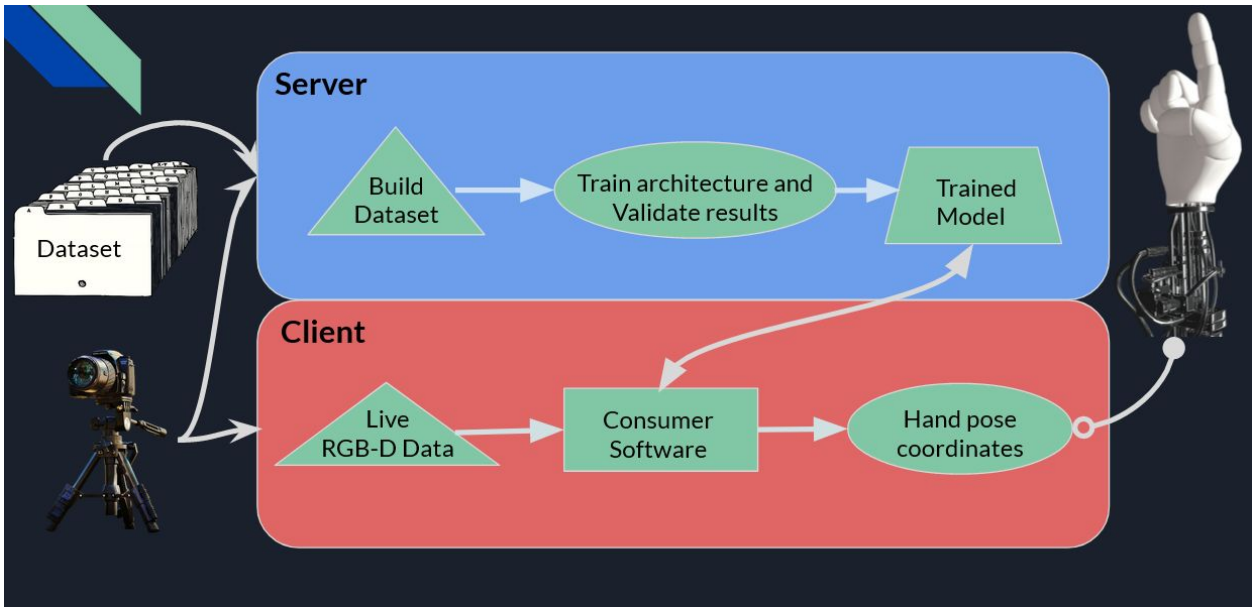
- Server: necessary for the training stage, Used to train the network before its deployment at the Client side.

- Client: The user of the software, will stream a live input of images of his hands using a RGB-D camera to the software in his computer which will output hand and finger poses.

- Simulation: a Unity based program that will receive as input the hand and joints positions from our software and present the movement in a virtual environment.



Block Diagram:
The server trains the neural network on a dataset which includes photos of hands and labeled coordinates for poses, The model can then be saved to the clients computer and fed inputs from a Live RGB-D feed and output coordinates to a third party software (Unity Game engine, Mechanic hands etc).



9. Use Cases

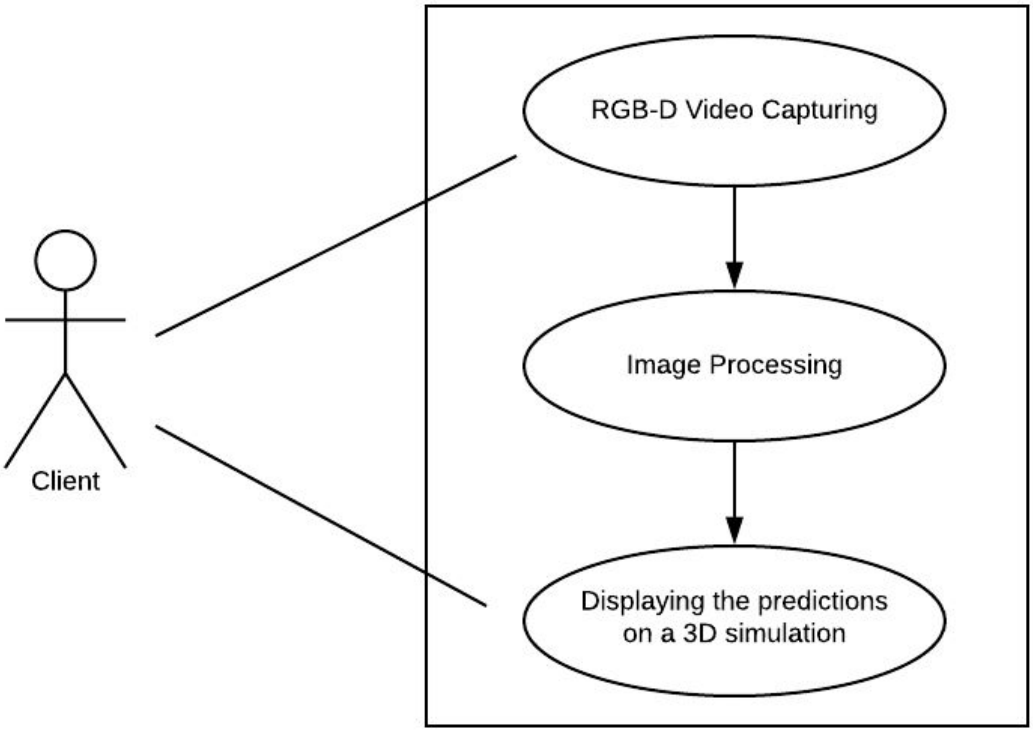
Identifying the Actors and Stakeholders:

The Clients - Actor - will be able to interact with virtual objects by moving their own hands while using only a single camera and no additional hardware (apart from a computing unit).

Medical Facilities - Stakeholder - will be able to install the software in operating rooms and transfer the very subtle and precise hand movements of their surgeons to robotic systems.
It can be proved vital in situations where the patient himself needs to be under strict quarantine and direct contact between him and the surgeon is not possible.

Video Game Developers - Stakeholder - will be able to expand the field of Virtual Reality based games by implementing ways to further interact with the environment and virtual objects inside the game.
With the new system the player will be able to make subtler and more precise moves without the need to gears or joysticks which could improve the accessibility of VR games to a much bigger market segment.

The Main Processes / Functions of the project:



RGB-D Video Capturing - The RGB-D camera will capture the client's hand and transfer it as a live inputstream of images for the algorithm.

Image Processing - The input steam will be preprocessed into a series of images which will be normalized and then transferred to the neural network.

Displaying the predictions on a 3D simulation - the output of the network will be a list of coordinates of the hand and its joints. The coordinates will be transferred to a Unity based program that will simulate the hand movement in a virtual environment.

10. Alternative Technologies

- **TensorFlow - A framework for machine learning**:
  This framework is similar to pytorch with Tensorflow's main difference with the usage of static graphs, while PyTorch uses dynamic graphs. Although TensorFlow allows faster performance due to that, this project requires experimentations with different architectures and Pytorch is better in that aspect. Tensorflow 2, offers dynamic graphs as well as a new parallel processing technology, which could allow a fast learning rate on an appropriate (parallel) network.

- **RGB Camera - Colored Image:**
  RGN camera captures images with three color channels and is more widely used. They are also cheaper and easy to find. However, existing techniques capturing and assessing poses perform significantly better with an additional depth channel.

- **Amd GPU - A graphic processor for image processing tasks:**
  Although AMD GPUs are compatible with machine learning tasks, the industry standard for such tasks are GPUs by Nvidia, and most of the popular libraries for these tasks are optimized for GPUs.

- **Nvidia TPU - Tensor Processing Unit:**
  a custom-built integrated circuit developed specifically for machine learning and tailored for TensorFlow, and in many use cases, it offers better performance than GPUs.

11. Tools and Means

11.1. Hardware and Software

Hardware :

- A computer with multi core CPU, and modern GPU compatible with most machine learning frameworks.
- RGB-D Camera in order to record images and insert them as an inputstream to the trained model.
- Arduino (sensor)

Software :

- Python 3
- Anaconda
- Pytorch
- Custom tools that we will design in order to create hand masks and annotations
- Arduino

11.2. Development and Execution

- Python 3
- OpenCV
- Anaconda
- Pytorch / Tensorflow2
- A Dataset comprises RGB-D images of hands and their corresponding labels.

11.3. Gaps Of Knowledge

- We are not yet familiar with the RGB-D camera and all of its features, so a learning process is required in order to use it in our project.

12. Project Products

12.1. General Description

1) SOW (Statement Of Work) -

2) PR (Progress Report) -

3) MR (Midterm Report) -

4) Project Journal (Final Report) -

5) Software (Alpha) - Software which receives a stream of RGB-D images of a hand as input, and outputs coordinates for the hand's joints and finger tips. The input for the software will first be preprocessed

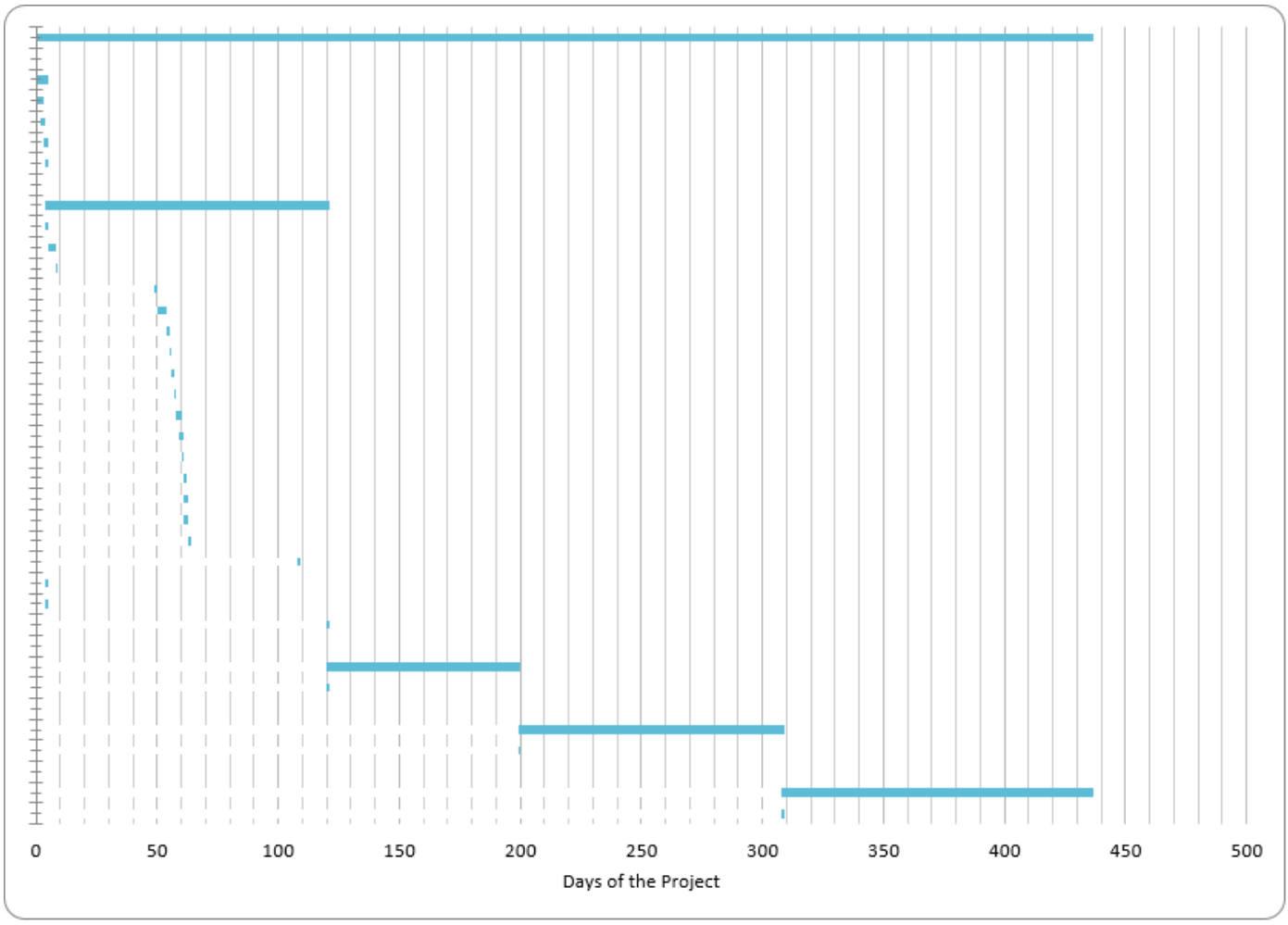
12.3. Project Life Span

Several architectures, loss functions and optimizations methods will be tested. The main measurement for the accuracy of the model is the average distance between the coordinates of the output and the labels of the test set.

13. Programme Of Work

TASK NAME	START DATE	END DATE	START ON DAY	DURATION (WORK DAYS)	TEAM MEMBER
Programme Of Work	15/03/2020	25/05/2021	0	437	
Selecting the Subject of the Project	15/03/2020	19/03/2020	0	5	
First team meeting and brainstorming	15/03/2020	17/03/2020	0	3	Nir & Daniel
Initial gathering of information	17/03/2020	18/03/2020	2	2	Nir & Daniel
Advisor Confirmation	18/03/2020	19/03/2020	3	2	Ehud Dayan
Submission of the proposed project	19/03/2020	19/03/2020	4	1	
SOW - Statement Of Work	19/03/2020	13/07/2020	4	117	
Team meeting	19/03/2020	19/03/2020	4	1	Nir & Daniel
Searching for papers and reviewing articles about the subject	20/03/2020	22/03/2020	5	3	Nir & Daniel
Objectives, goals and measurements	23/03/2020	23/03/2020	8	1	Nir & Daniel
Initial Elevator Pitch Presentation	03/05/2020	03/05/2020	49	1	Nir & Daniel
Literature Review	04/05/2020	07/05/2020	50	4	Nir & Daniel
Introduction	08/05/2020	08/05/2020	54	1	Nir & Daniel
Current Solutions	09/05/2020	09/05/2020	55	1	Nir & Daniel
Competitors Comparison	10/05/2020	10/05/2020	56	1	Nir & Daniel
Initial Market Survey	11/05/2020	11/05/2020	57	1	Nir & Daniel
System Requirements	12/05/2020	12/05/2020	58	1	Nir & Daniel
Substitute Technologies	13/05/2020	13/05/2020	59	1	Nir & Daniel
Tools and Means	14/05/2020	14/05/2020	60	1	Nir & Daniel
Gaps of Knowledge	15/05/2020	15/05/2020	61	1	Nir & Daniel
Products: General Description	15/05/2020	16/05/2020	61	2	Nir & Daniel
Risk Management	15/05/2020	16/05/2020	61	2	Nir & Daniel
SOW Presentation	17/05/2020	17/05/2020	63	1	Nir & Daniel
Sending the draft to the Project Advisor	01/07/2020	01/07/2020	108	1	Nir & Daniel

Correcting the draft					Nir & Daniel
Advisor Confirmation					Ehud Dayan
Submission of the project's SOW	13/07/2020	13/07/2020	120	1	Nir & Daniel
PR - Progress Report	13/07/2020	30/09/2020	120	80	
Submission of the project's PR					Nir & Daniel
MR - Midterm Report	30/09/2020	17/01/2021	199	110	
Submission of the project's MR					Nir & Daniel
Project Journal - Final Report	17/01/2021	25/05/2021	308	129	
Submission of the project's Final Report					Nir & Daniel



14. Gaps

14.1. Gaps Of Knowledge

The gaps of knowledge that we had to deal with and resolve were as follows:

- Lack of background knowledge regarding the science behind Human-Computer Interaction
- Lack of experience with machine learning processing in real time
- Lack of knowledge in the field of game engines and Unity in specific.
- We had basic knowledge and understating of the VR technology but primarily from a consumer side. A further study was required in order to get the optimal results.

We've managed to bridge the gaps of knowledge by searching for articles and acquiring relevant information about each subject.

- In addition we lacked knowledge in the field of Computer Vision, but we've managed to overcome it by completing the relevant course and practicing it in Kaggle.

14.2. Equipment

Hardware:

- A computer with multi core CPU, and modern GPU compatible with most machine learning frameworks.
- RGB-D Camera in order to record images and insert them as an inputstream to the trained model.

Software:

- Python 3 - Runs most machine learning frameworks
- Anaconda - popular version and package manager for python.
- Python libraries for images and videos processing.
- Pytorch / Tensorflow2 for building and operating Deep Learning networks.

Additional:

- A Dataset comprises RGB-D images of hands and their corresponding labels.

15. Risk Management

Possible risks:

- Low quality or too small of a dataset
- Bad choice in model - architecture
- Bad choices in HyperParameters (loss function, learning rate)
- Results with low performance under different conditions, (different lightings, backgrounds)

16. Bibliography

- “Efficient Hand Pose Estimation from a Single Depth Image”[1]
- “Rule Of Thumb: Deep derotation for improved fingertip detection”[2]
- “First-Person Hand Action Benchmark with RGB-D Videos and 3D Hand Pose Annotations”[3]

[1] https://www.cv-foundation.org/openaccess/content_iccv_2013/papers/Xu_Efficient_Hand_Pose_2013_ICCV_paper.pdf

[2] https://arxiv.org/pdf/1507.05726.pdf

[3] https://arxiv.org/pdf/1704.02463.pdf