

# AI based Electronic Component Identifier

**Student:** Violet Concordia

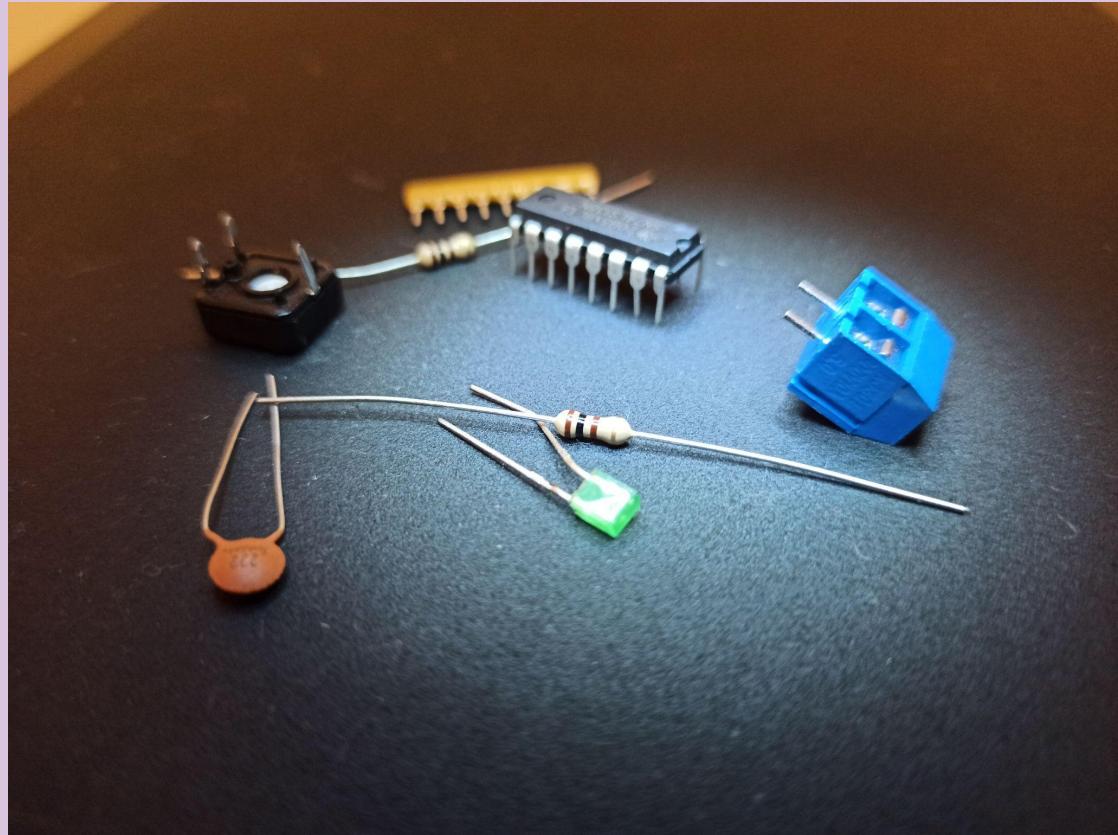
**Student number:** B00125142

**Supervisor:** Benjamin Toland

**Course ID:** TU807

# Object detection

- Important in
  - Security
    - Notifying concerns
      - People
      - Animals
      - Flora
  - Production
    - Discarding defects
      - Damage
  - Analysis
    - Quality inspection
      - Scratches
      - Spots
    - Classification
      - Resistor
      - Capacitor
    - Total and class count



```
11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11  
11 11 11 11 11 11 11 10 10 11 11 11 11 11 11 11 10 10 11 11 11 11 11 11  
11 11 11 11 11 11 11 11 10 11 11 11 10 11 11 11 10 11 11 11 11 11 11 11  
11 11 11 11 11 11 11 11 11 10 11 10 11 11 11 10 11 11 11 11 11 11 11 11  
11 11 00 00 00 11 11 11 11 10 11 10 11 11 11 11 11 11 11 00 00 00 11 11  
11 00 01 01 01 00 00 11 11 10 11 10 11 11 11 00 00 01 01 01 00 11  
11 00 01 01 01 01 01 00 11 11 10 11 11 11 00 01 01 01 01 01 00 11  
11 00 01 01 01 01 01 01 00 10 10 10 00 01 01 01 01 01 01 01 00 11  
11 11 00 01 01 01 01 01 01 10 10 10 01 01 01 01 01 01 01 00 11 11  
11 11 00 01 01 01 01 01 01 10 10 10 01 01 01 01 01 01 01 00 11 11  
11 11 11 11 00 01 01 01 01 10 10 10 01 01 00 00 00 11 11 11 11  
11 11 11 11 00 00 00 00 11 11 10 11 11 10 00 00 01 01 01 00 11 11  
11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11  
11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11  
11 11 11 11 11 11 10 10 11 11 11 11 11 11 10 10 11 11 11 11 11 11  
11 11 11 11 11 11 11 11 11 10 11 11 11 10 11 11 11 11 11 11 11 11  
11 11 11 11 11 11 11 11 11 11 10 11 10 11 11 11 11 11 11 11 11 11  
11 11 00 00 00 11 11 11 11 10 11 10 11 11 11 11 11 00 00 00 11 11  
11 00 01 01 01 00 00 11 11 10 11 10 11 11 00 00 01 01 01 00 11  
11 00 01 01 01 01 00 11 11 10 11 11 11 00 01 01 01 01 00 11  
11 00 01 01 01 01 01 00 10 10 10 00 01 01 01 01 01 01 00 11  
11 11 00 01 01 01 01 01 10 10 10 01 01 01 01 01 01 01 00 11 11  
11 11 00 01 01 01 01 01 10 10 10 01 01 01 01 01 01 01 00 11 11  
11 11 11 00 01 01 01 01 10 10 10 01 01 01 01 01 01 00 11 11 11  
11 11 11 11 00 00 00 01 01 10 10 10 01 01 00 00 00 11 11 11 11  
11 11 11 00 01 01 01 00 00 10 10 10 00 00 01 01 01 01 00 11 11 11  
11 11 00 01 01 01 01 01 01 10 10 10 01 01 01 01 01 01 01 00 11 11 11  
11 11 00 01 01 01 01 01 00 10 10 10 00 01 01 01 01 01 01 00 11 11 11  
11 11 00 01 01 01 01 00 11 11 10 11 11 11 00 01 01 01 01 00 11 11 11  
11 11 11 00 00 00 00 11 11 11 10 11 11 11 00 00 00 00 11 11 11 11  
11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11
```

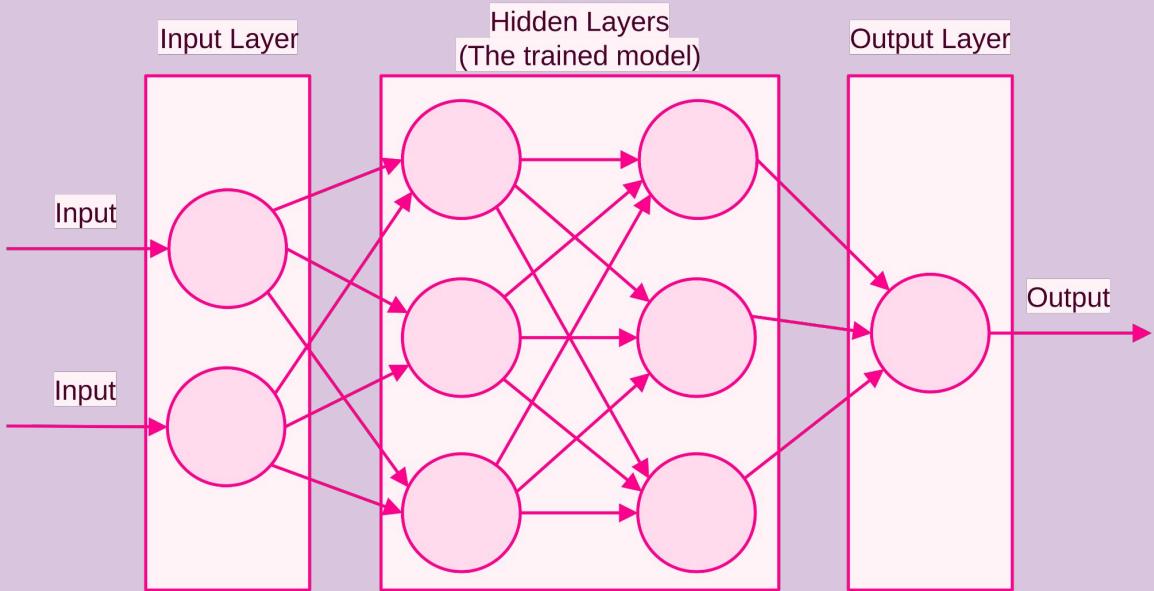
# The problem

- **Object detection from images:**
  - Natural to **intelligent** creatures.
    - Designed for **object detection** through evolution.
      - Second nature.
  - **Binary data to computers.**
    - Has no concept of **object, image, or color.**
      - Everything is processed the same.

# The solution

- **Neural Networks**
  - Complex networks of **neurons**.
  - Each **neuron** holds weight for specific input bias.
    - Trained against certain **datasets**, in hope of recreating the function.
  - The **output** is processed by the **neurons**.
- In image **object detection**, input is each individual pixel.

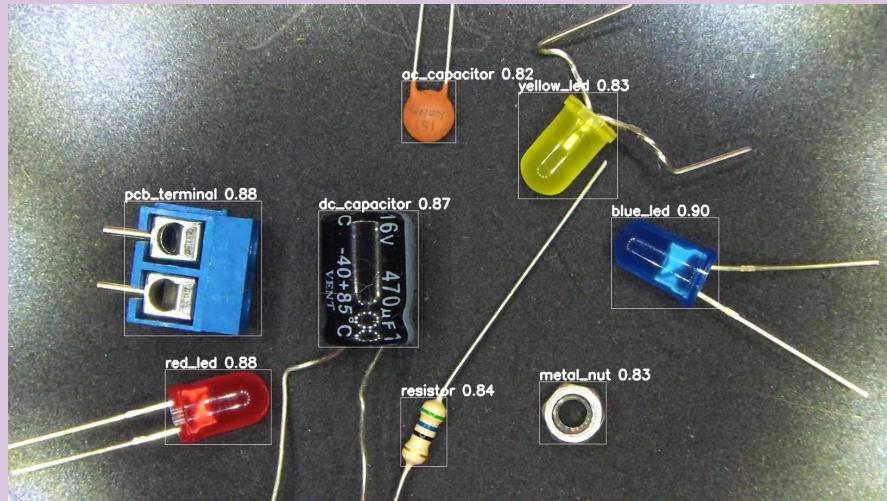
## Simple Neural Network Diagram



# AI based Electronic Component Identification

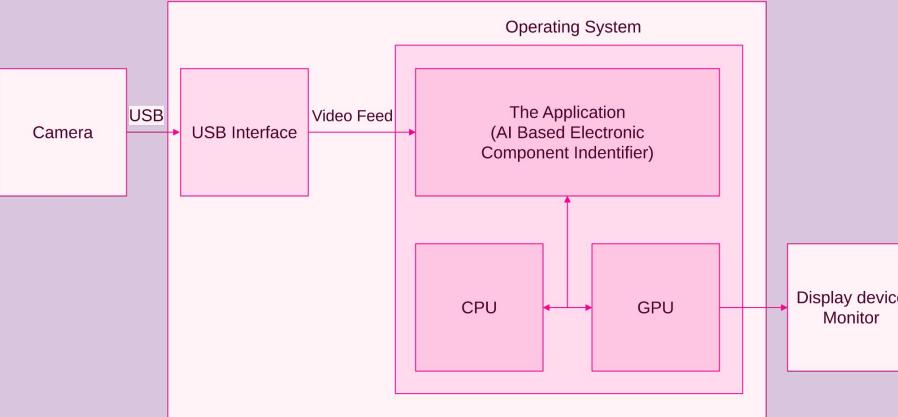
## GUI display

- **Identifies:**
  - **Position** (Bounding Box on a live display).
  - **Class name.**
    - Resistor
    - Capacitor
  - **Confidence** (Percentage).
  - **Quantity.**
  - Potential **extra information:**
    - Resistors - **Color code** to Ohms.
    - Capacitors - **Number code** to farad.
    - IC - **Pin count, text on the IC if visible.**

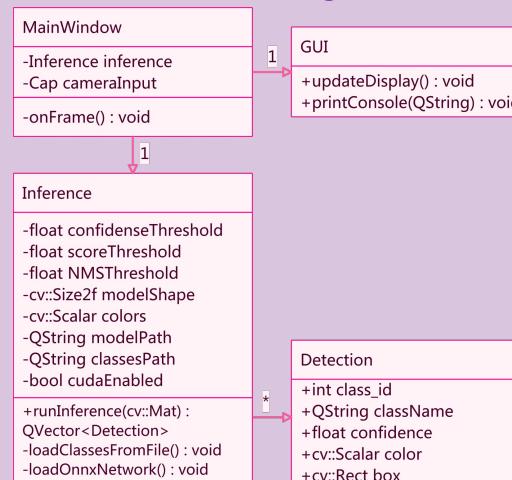


# System Block Diagram

Machine running the Application



## Class Diagram



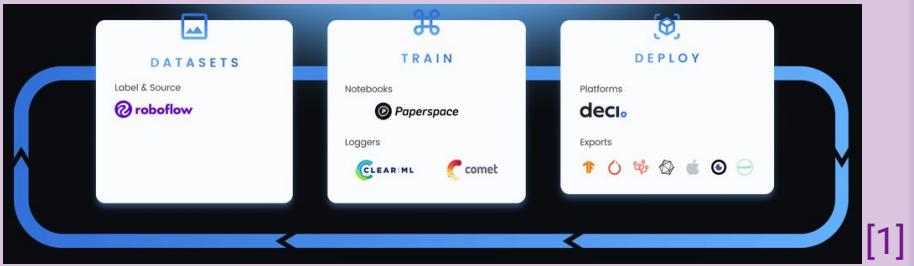
# Concept Diagrams

- **System Block Diagram**

- **Camera** is External.
- Communication through **USB**.
- The application utilises both **CPU** and **GPU**.

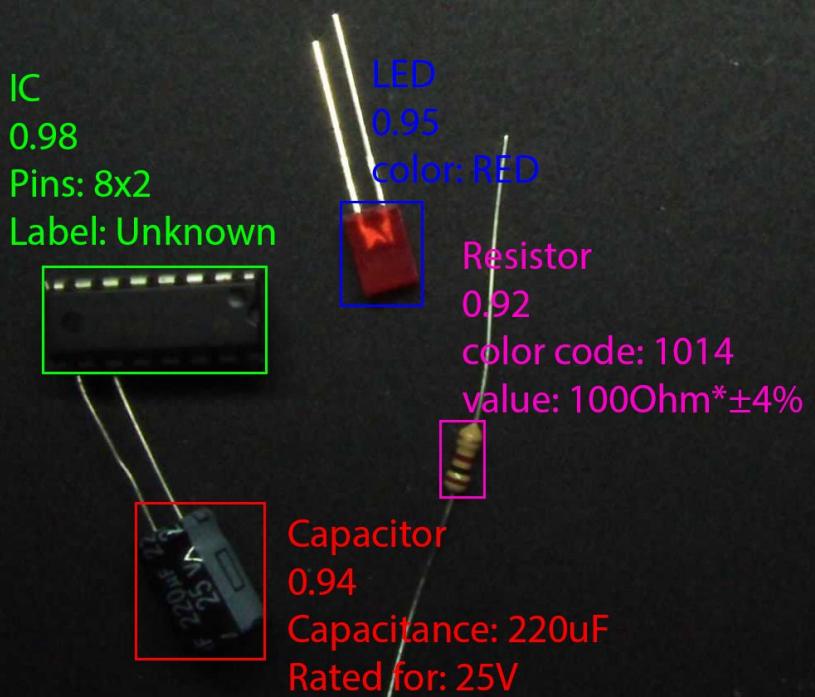
- **Class Diagram**

- MainWindow is the controller.
- **Inference** is responsible for **object detection**.
- **GUI** is the output.



# Model Training

- Trained on over **3000** images
  - Taken on the **rig**.
  - Manually **labeled**.
- **Each picture** should have some **variation** in conditions.
  - Position
  - Rotation
  - Lighting



Note: For now, this was achieved through image manipulation.

# Post-Processing

Additional processing of the **output** provided by the **inference**.

- **Identification** of (For those that apply)
  - **Labels** using **Text Recognition**
  - **Color/number codes**
  - **Pin count**
  - **Color** (LED)

# Discussion

- In comparison to **algorithms**, and sometimes even **humans - inference** is capable of extracting an **Extremely high** level of detail from an input image.
  - As **post-processing** is **algorithmic**, it is expected to be **considerably less accurate**.
    - **Missed letters** must depend on **auto-correction**, which is another **algorithm**.
    - **Color shift** may result in a **wrong color code read**.
- **Gathered** over **3000 labeled images**.
  - The **training** of the current **model** for **300 epochs** takes over **10 hours**.

# Conclusion

- Sufficient amount of **data** has now been **gathered**, yielding **impressive results**.
  - While the **project** is mostly **software focused**, a **high quality rig setup** was essential in gathering a **high quality dataset**.
  - This **concludes** the **gathering** of additional **data** to prioritise **other milestones**.

# References

- [1] YoloV5 <https://github.com/ultralytics/yolov5>, accessed on 6th of November, 2022



The end

Any questions?