mind, a computer doing the same thing is most certainly non-trivial. Teaching a computer to recognise certain objects is notably similar to teaching a baby to apply names to certain objects, except in this case you must first tell the baby how to learn. To teach a computer how to recognise an object using machine learning, a huge dataset is initially needed to 'train' the algorithm about what represents a certain object.

This project will attempt to build a web-based game, where a player will race against one or more types of image classification methods to guess the content of an image. The game should implement some kind of progress tracking, where the player can judge their performance history. This could be done using some kind of database software, to keep a history of the users scores, possibly linked to a username in permeant storage on a server, or just in temporary storage on the local machine. Furthermore, some kind of difficulty system should be implemented, so as the player progresses through the game, it should get more challenging as time goes on. The challenge of difficulty is complicated as it will entirely depend on how good the image classification method is. If the algorithm can guess the images consistently more accurately than a human, which has been possible in a few types of algorithms **(He et al., 2015)** then difficulty could be introduced by implementing a handicap for the algorithm to begin with, and slowly removing it. On the other hand, if the player is consistently better than the algorithm, then a handicap would need to be slowly introduced for the player as they progress. The requirement of a handicap and what kind of handicap is needed depends on the image classification algorithm used, so research will be needed to find the best algorithm for the fairest gameplay. If there is time, more than one algorithm could be implemented as another type of difficulty tier. An assessment will need be made of each algorithm's ability to run on a local machine (more specifically, in a browser), and their viability for competing with a player. For algorithms that will run locally, a dataset may be needed to 'train' the algorithm of image categories. A widely available and non-copyrighted set such as Caltech 101 **(Li Fei-Fei, Fergus and Perona, 2006)** could be used as this set has been used as a benchmark for many classification algorithms, and has plenty of data available for comparisons of algorithm success rates etc.

1.3 Aims

1. Develop a web-based game with user progress tracking, and multiple difficulty tiers
2. Explore the differences between humans and algorithms when trying to recognise objects in images

1.4 Objectives

1. Review existing literature & assess viable options of image classification algorithms
2. Research languages best-suited for a web-based game
3. Research UI frameworks and back-end frameworks for use in a web-based game
4. Design GUI mock-up highlighting the main areas of the application and how it is controlled
5. Create prototype game with rudimentary UI, to test and develop gameplay mechanics (such as difficulty)
6. Implement mock-up GUI design
7. Perform analysis of win/loss rates and implement difficulty levels
8. Document progress, findings and procedure in report