|  |
| --- |
| Investigation of Visual Bias in Generative AI |
| Jerome Agius |
| Supervisor: Dr Dylan Seychell |
| May 2024 |
| Submitted in partial fulfilment of the requirements for the degree of Bachelor of Science in Information Technology (Hons) (Artificial Intelligence). |



Abstract (max 300 words)

In the ever-evolving world of Artificial Intelligence (AI), text-to-image generators, such as Stable Diffusion, Dall-E-3 and Midjourney revolutionise creativity, but raise concerns regarding bias in generated images, particularly those depicting people. Bias can also present itself in the training datasets used to build these models. This thesis investigated this issue by comparing and analysing the inherent bias within these models and popular training datasets.

The research approach revolved around the retrieval/generation of images coinciding with the terms *person, doctor,* and *nurse*. The latter two terms were used to leverage real-world biases throughout the bias identification process thus, exposing how each model deals with this innate bias. Following this, image subsets extracted from the datasets were human annotated to expose inherent bias within the DeepFace implementation which was used to extract the image features.

The presence of bias was determined based on a set of metrics, which consisted of gender, race, age and emotion distributions, metric groupings, and person prominence. These findings expose add overview of the results and conclusion reached as well as any anti-bias measures identified.

This research sheds light on the pervasiveness of bias in generative AI, highlighting the urgent need for proactive mitigation strategies. Our findings contribute to understanding bias and developing fairer models and datasets. Future work could explore advanced anti-bias techniques and broader societal implications of biased image generation.

Acknowledgements

I would like to thank my supervisor Dr Dylan Seychell for guiding me throughout the process of this final year project and aiding me throughout the various challenged encountered. I would also like to thank my parents, Reno and Graziella, and my brother Julian for their continuous support.

Contents

[Abstract (Style: Headings) ii](#_Toc120111054)

[Acknowledgements iii](#_Toc120111055)

[Contents iv](#_Toc120111056)

[List of Figures vi](#_Toc120111057)

[List of Tables vii](#_Toc120111058)

[List of Abbreviations viii](#_Toc120111059)

[1 Introduction (Style: Heading 1) 1](#_Toc120111060)

[1.1 How to Use this Template (Style: Heading 2) 1](#_Toc120111061)

[1.2 Using Styles 3](#_Toc120111062)

[1.3 Line Spacing for Headings **Error! Bookmark not defined.**](#_Toc120111063)

[1.4 Inserting Equations **Error! Bookmark not defined.**](#_Toc120111064)

[1.5 Automatically updating of Cross-Referencing **Error! Bookmark not defined.**](#_Toc120111065)

[1.5.1 Using Styles Correctly (Style: Heading 3) **Error! Bookmark not defined.**](#_Toc120111066)

[1.5.1.1 Sub-Titles (Style: Heading 4) **Error! Bookmark not defined.**](#_Toc120111067)

[1.6 Some Hints on the Dissertation Writing Style **Error! Bookmark not defined.**](#_Toc120111068)

[1.7 Some Hints on using *Word* **Error! Bookmark not defined.**](#_Toc120111069)

[1.8 Avoiding Plagiarism **Error! Bookmark not defined.**](#_Toc120111070)

[1.9 How to Write the Introduction **Error! Bookmark not defined.**](#_Toc120111071)

[2 Background and Literature Review 4](#_Toc120111072)

[2.1 References **Error! Bookmark not defined.**](#_Toc120111073)

[2.1.1 *RefWorks* **Error! Bookmark not defined.**](#_Toc120111074)

[2.1.2 Formatting the Reference Section **Error! Bookmark not defined.**](#_Toc120111075)

[2.1.3 Examples of Different Types of References **Error! Bookmark not defined.**](#_Toc120111076)

[2.2 Writing the Background and Literature Review Chapter 6](#_Toc120111077)

[3 Specification and Design 8](#_Toc120111078)

[3.1 Further use of Captions 8](#_Toc120111079)

[3.1.1 Tables 8](#_Toc120111080)

[3.1.2 Lemmas, Theorems, Corollaries etc. 9](#_Toc120111081)

[3.2 Chapter Headers 11](#_Toc120111082)

[3.3 Printing the Dissertation on Both Sides 12](#_Toc120111083)

[3.4 Writing the Specification and Design Chapter 15](#_Toc120111084)

[4 Implementation 16](#_Toc120111085)

[4.1 Writing the Implementation Chapter 16](#_Toc120111086)

[5 Testing and/or Evaluation 17](#_Toc120111087)

[5.1 Writing the Evaluation and/or Evaluation Chapter 17](#_Toc120111088)

[6 Future Work 18](#_Toc120111089)

[6.1 Writing the Future Work Chapter 18](#_Toc120111090)

[7 Conclusions 19](#_Toc120111091)

[7.1 Writing the Conclusions Chapter 19](#_Toc120111092)

[References 20](#_Toc120111093)

[Bibliography 21](#_Toc120111094)

[Appendix A When to Use Appendices (Style: Appendix Heading 1) 22](#_Toc120111095)

[A.1 Appendices Headings (Style: Appendix Heading 2) 22](#_Toc120111096)

[A.1.1 Appendix Sub-Headings (Style: Appendix Heading 3) 22](#_Toc120111097)

[A.2 What to put in Appendices 22](#_Toc120111098)

[Appendix B Version Information 23](#_Toc120111099)

[B.1 Copyright Information 23](#_Toc120111100)

[B.2 Suggestions 23](#_Toc120111101)

List of Figures

[Figure 1.1 Cross-referencing (Style: Caption Figure). **Error! Bookmark not defined.**](#_Toc120084584)

[Figure 1.2 Inserting a caption to a figure. **Error! Bookmark not defined.**](#_Toc120084585)

[Figure 1.3 Figure example, a rectangle and a triangle. **Error! Bookmark not defined.**](#_Toc120084586)

[Figure 1.4 Keeping paragraphs or lines on the same page. **Error! Bookmark not defined.**](#_Toc120084587)

[Figure 2.1 *RefWorks Citation Manager* settings options. **Error! Bookmark not defined.**](#_Toc120084588)

[Figure 2.2 Inserting a citation using *RefWorks*. **Error! Bookmark not defined.**](#_Toc120084589)

[Figure 2.3 Content Control Box containing the references. **Error! Bookmark not defined.**](#_Toc120084590)

[Figure 2.4 Correctly formatting the Reference section. **Error! Bookmark not defined.**](#_Toc120084591)

[Figure 3.1 Converting document to printing on both sides. 14](#_Toc120084592)

[Figure 3.2 Inserting chapter headers. 15](#_Toc120084593)

List of Tables

[Table 3.1 Simple table example. 8](#_Toc120084594)

[Table 3.2 A more complicated table example. 9](#_Toc120084595)

List of Abbreviations

FYP Final year project (Style: Abbreviations)

AI Artificial Intelligence

GAN Generative Adversarial Network

VAE Variational Autoencoder

CLIP Contrastive Language-Image Pre-training

UNET U-shaped encoder-decoder network architecture

ResNet Residual Neural Network

Note that the List of Abbreviations should be sorted on the acronym list.

The entries in the List of Abbreviations should be assigned the Abbreviations style.

# Introduction

## Problem Definition

In recent years, the field of Generative AI has experienced remarkable advancements in visual content generation, with a primary focus on images. Notably, generative models such as Midjourney, DALL-E and Stable Diffusion have been at the forefront of this progress \cite{midjourney, dall-e-2, stable-diffusion-online}, by providing users with the capability to generate numerous images through the use of a simple text prompt.

However, the generation of visual content brings to the forefront a variety of critical issues such as lack of control over output, over fitting as well as privacy and ethical concerns \cite{Controllable-Generative-Adversarial-Network, GAN-Privacy-Ethics-Concerns}.

This study focuses on a particular issue, that of bias. Bias in relation to visual AI systems tends to refer to cases in which systems showcase prejudice in relation to gender and race \cite{Bias-Gender-Race}. Several instances exist in which this prejudice led to negative consequences in relation to recidivism scoring \cite{COMPASS-situation-racial-bias}, online advertisement \cite{Discrimination-in-Online-Ad-Delivery}, facial recognition \cite{Facial-Recognition-Negative-Consequnces}, and credit scoring \cite{Credit-Scoring-Negative-Consequnces}.

Bias serves to affect a large majority of computer vision systems such as classification algorithms, face recognition systems, object detectors and many more \cite{RefWorks:RefID:30-fabbrizzi2022survey}. To address this problem tools can be created which aid in the identification of bias, these are crucial as bias is not attributed to a singular cause rather a variety of factors varying from the composition of the dataset and the framing of images to the characteristics of the latent space employed during the generative process \cite{RefWorks:RefID:30-fabbrizzi2022survey}.

Tools such as this already exist, a prime example is the REVISE implementation which given an annotated dataset can provide object-based, person-based and geography-based insights on the presence of bias \cite{revisetool\_eccv}. However, such systems tend to be cumbersome to set-up and utilise. The initial aim of this study was to detect if bias is present in traditionally gender biased prompts such as doctor and nurse by looking at the prompt associated images of the LAION-5B dataset as well as generated images from the Stable Diffusion model to detect any forms of bias with a focus on gender and race. However, due to recent proceedings with the LAION-5B dataset, wherein access to said dataset was revoked the aim of the study was shifted \cite{<https://cointelegraph.com/news/laion-5b-ai-data-set-removed-child-sexual-abuse-material>}. Thus, this study will attempt to outline the presence of bias in several popular datasets as opposed to the LAION-5B dataset whilst also considering various generative models these being Stable Diffusion, DALL-E and Midjourney. This study also aims to develop a simple to use python notebook which will facilitate image feature extraction and metric visualisation to allow individuals to easily detect bias.

## Motivation

The motivation behind this research stems from the growing importance of addressing bias in artificial intelligence (AI) systems, particularly within the realm of generative models and visual datasets. As AI technologies continue to play an increasingly integral role in shaping various aspects of our lives, understanding and mitigating biases becomes imperative. The {add dataset list} datasets and Midjourney, DALL-E and Stable Diffusion models serve as focal points for this study, representing key components in the landscape of generative AI. By investigating and uncovering biases present in these specific entities, this research aims to contribute valuable insights to the broader discourse on ethical AI development. The implications of biased AI systems are far-reaching, with potential consequences in areas such as image generation, facial recognition, and algorithmic decision-making. Through a meticulous examination of biases, this study strives to not only enhance our understanding of the challenges inherent in generative models but also to pave the way for more ethical and unbiased AI systems in the future.

## Aims and Objectives

The aim of this study as outlined above is to determine the presence of bias in popular training datasets and generative AI. This aim will be achieved via the following set of objectives:

1. Analyse bias-associated prompts and determine an optimal feature extraction model. Determine the requirements needed to select appropriate human annotators for valid annotations. Resulting in an optimal prompt structure by which images can be generated using any label and model as well as denoting the requirements for valid human annotations.
2. Generate images using pre-defined prompts containing the *Doctor* and *Nurse* terms with Midjourney, DALL-E, and Stable Diffusion. Extract the main image features (gender, race, age) from both generated and training dataset images, including a human-annotated training dataset subset for bias detection in the feature extraction model used. This will result in three annotated image sets: generated, training data, and human-annotated training data alongside the identification of inherent bias of the feature extractor.
3. Analyse the extracted features consisting of gender, race, age, emotion distributions, and overall person prominence across data groups. Visualize these metrics to aid in identifying relationships between the data and drawing conclusions.
4. Through expert interviews and qualitative analysis regarding the visualised metrics, uncover relationships within the data to identify the optimal training dataset and model in terms of lack of bias, while revealing common bias manifestations in training datasets and models.

## Document Structure

Add this section later once the document starts to take more shape.

# 

# Background

This chapter provides a foundation of knowledge required for understanding the techniques employed within the developed system for bias detection in both visual datasets and generative models by introducing the relevant background concepts and techniques.

The chapter is divided into three subsections covering *prompting*, *CLIP*, diffusion models, *face recognition, facial analysis,* and *image bias,* going over a variety of relevant research and challenges associated with each section. Furthermore, the chapter outlines how each section fits into this research paper*.*

## Prompting

Prompting in the context of AI models can be defined as the act of providing the model with instructions that guide the generation process of text, code, images, and other varied outputs. These instructions can take various forms, the most common of which being text, code, and images. Given that this research paper concerns itself with text-to-image generation and the bias therein only text inputs and image outputs are considered.

Although on the surface prompting appears quite straight forward it brings with it a variety of challenges mainly in relation to retrieving relevant images. This challenge is closely tied with identifying the right prompt, which is a non-trivial task as it not only takes a significant amount of time but minor changes to the prompt could result in a huge impact on performance \cite{coop}. This is where prompt engineering comes into play which involves altering the prompt in a variety of ways such as altering its length and wording used to affectively depict the required output as opposed to merely specifying the desired image \cite{[Prompt engineering - OpenAI API](https://platform.openai.com/docs/guides/prompt-engineering/six-strategies-for-getting-better-results)}. This process can also be enhanced through automated prompt engineering however this was beyond the scope of this research paper.

## CLIP

CLIP is a multimodal vision and language model developed by OpenAI which was trained on 400 million image-text pairs collected from publicly available sources on the Internet in an attempt to cover as broad a set of visual content as possible \cite{clip-paper}. Contrary to traditional models which predict a fixed label for images, CLIP adopts a contrastive learning approach which allows it to learn the relationship between image and text pairs. This approach allows CLIP to determine the best image-text pairs for any possible use case. CLIP further differs from traditional image classifiers as it utilises zero-shot learning, a technique whereby a model is able to generalise to unseen classes without the need for training, thus allowing it to deal with never-before-seen images and classes \cite{zero-shot-learning}.

CLIP is relevant to this paper as it was used to retrieve images from the LAION-5B dataset via the clip-retrieval library by converting the text query to a CLIP embedding, then using that embedding to query a k-nearest neighbour index of clip image embeddings \cite{LAION5BClipSearch, clip-retrieval}. This model was used to retrieve images from the LAION-5B dataset prior to it being taken down. Additionally, CLIP is also used within both Stable Diffusion and DALL-E as a text encoder to generate text-embeddings required by said models to generate correct images \cite{stable-diffusion-clip-reference, dall-e-clip-reference}. It is undisclosed if Midjourney also utilises CLIP in a similar manner or at all.

## Diffusion Models

Generative models encompass a variety of different approaches, including GANs, VAEs, and diffusion models. The latter offers several advantages over its counterparts. Unlike GANs, diffusion models excel in both training stability and diverse image generation, avoiding the pitfalls that often plague GANs. Additionally, they bypass the surrogate loss issue inherent in VAEs. This allows diffusion models to achieve superior performance and efficiency. The models considered in this paper all fall under the diffusion category \cite{dall-e-3-paper, stable-diffusion-paper, midjourney-pickfu-article}.

Diffusion models are traditionally composed of two steps, these being the forward and reverse diffusion processes. The forward diffusion process starts off with a clear image and slowly but gradually adds gaussian noise to it with every passing step. This is repeated until the input image no longer resembles the original input, these series of steps can be seen in Figure 2.1.

A diagram of a person's face

Description automatically generated

Figure 2.1 Forward diffusion process \cite{diffusion-models-explained}

The reverse diffusion process employs a noise prediction model which iteratively refines the noisy forward pass image towards a clear output. This process resembles the inverse of the forward pass as can be seen in Figure 2.2. The core component of this de-noising process is the UNet architecture, these are convolutional [neural networks](https://aws.amazon.com/what-is/neural-network/) originally developed for image segmentation in biomedicine. In particular, Stable diffusion adopts the ResNet model developed for computer vision. During each step of the iterative process, the noise predictor estimates the noise component present in the latent space representation of the image. This estimated noise is then subtracted, effectively denoising the image. This cycle repeats for a predefined number of steps, progressively removing noise and enhancing the image's detail. Notably, the noise predictor can be guided by conditioning prompts, influencing the final outcome and directing the image generation process towards specific themes or styles \cite{stablediffusion-process-explained}. The de-noising process can be seen in Figure 2.2.

A diagram of a mathematical equation

Description automatically generated

Figure 2.2 Reverse diffusion process \cite{diffusion-models-explained}

Further analysis of the Stable Diffusion model showcases that it utilises a variant of the diffusion model architecture known as the latent diffusion model. This model differs from the traditional diffusion model as it tackles the denoising stage within a compressed representation of the image, termed the "latent space," as opposed to the pixel space. This strategy offers significant computational advantages. Consider a standard 512x512 colour image, boasting a staggering 786,432 possible values. Contrastingly, Stable Diffusion operates on a compressed representation containing only 16,384 values, reducing its size by a factor of 48. This dramatic compression translates to tangible benefits, including vastly reduced processing demands, enhanced performance, and improved overall efficiency \cite{stablediffusion-process-explained, stable-diffusion-paper}.

## Face Recognition

Face recognition can be defined as a three-step process consisting of *Face detection*, *Feature extraction* and *Face recognition* where the input is always an image or video, and the output is the identification or verification of the image or video subjects \cite{face-recognition-pipeline, face-recognition-book}. Face detection is defined as the process by which image regions depicting faces are located and extracted, this has a variety of use cases like face tracking, pose estimation and compression. Feature extraction involves the retrieval of facial features from the data, which can be human relevant or not. These tend to include feature such as face regions, variations, angles, and measures. Feature extraction has a variety of use cases including facial feature tracking and emotion recognition. Face recognition utilises the outputs from the prior steps in conjunction with comparison methods, classification algorithms and an accuracy measure to recognise faces \cite{face-recognition}.

Although the pipeline is composed of three steps, there are cases in which Face detection is not carried out, particularly in instances where the images only contain the subjects face. This is not the case for the pipeline implemented in this paper as the images considered did not conform to this requirement. Furthermore, face detection must deal with several challenges \cite{face-recognition-book, face-recognition, face-recognition-2, face-recognition-3}:

* Pose variation – Large pose variation can severely decrease the performance of face detection algorithms. The obtain ideal results images should contain subjects forward facing.
* Feature occlusion – The obstruction of facial features can also decrease performance. This is usually caused by the presence of beards, glasses and other clothing items; however, faces can also be partially covered by objects or other faces.
* Facial expression – Different facial gestures can cause facial features to vary, thereby affecting facial detection.
* Imaging conditions – Image quality is a major factor in facial detection, this can be affected by the lighting conditions and image size which are determined by the camera and varying Ambiental conditions.

Feature extraction unlike face detection is crucial to the pipeline and it involves the extraction of a variety of differing features which depend on the feature extraction model used. These include colour-based, spatial, textural, geometric, and deep learning features. The type of features extracted vary based on the use case as colour-based features tend to see usage in image segmentation and retrieval, spatial features in object detection and image classification, textural features in texture analysis and material classification, geometric features in facial analysis and 3D face reconstruction and deep learning features in face recognition, object detection and image generation.

## Facial Analysis

Facial Analysis extends Face Recognition as it involves extracting relevant features from the detected faces. These can range from gender, age, emotion, race and more based on the implementation. This involves the use of a model trained to carry out such a task. In this paper's case three APIs were taken into consideration to facilitate Face Recognition and Analysis these being DeepFace, Inferdo and BetaFace \cite{deepFace-ref-1, deepFace-ref-2, inferdo, betaFace}. This was done as these three APIs not only had fair pricing but facilitated easy implementation with relatively accurate results.

## Image Bias

Bias is subjective / Various types

No set metric (mention REVISE or leave it for later)

Various measures

Hard to determine

## Chapter Summary

## Writing the Background and Literature Review Chapter

The purpose of the background section is to provide the reader with information that they cannot be expected to know but which they will need in order to fully understand and appreciate the rest of the project.

This section may describe such things as:

* the wider context of the project;
* the anticipated benefits of the system;
* the likely users of the system;
* any theory associated with the project;
* the software/hardware development method(s) used;
* any special diagramming conventions used;
* existing software (or hardware) that is relevant to the system;
* etc.

Since projects will likely include different kinds of theory, programming language choices, compilers, software/hardware components, APIs, development boards, IC technologies, one cannot always assume that the reader will be familiar with the details of all of them. The student should therefore explain concepts and use references to guide the reader.

The literature review component of the report should include:

* A study in the area of interested, highlighting the strengths and weaknesses of existing methods.
* A review of the state-of-the-art published material in the area.
* A critical analysis of exiting material and methods.
* An explanation showing why the literature chosen to review is relevant to the FYP.

# Specification and Design

## Further use of Captions

We have already seen the use of captions for figures and equations. Recall that this will allow automatic cross-referencing within *Word*. In addition to figures, you can also use captions to number and cross-reference other items in your dissertation.

### Tables

Tables 3.1 and 3.2 show two example tables. Notice that in the case of tables, the table caption needs to be placed above the table and assigned the *Caption Table* style. Also note that *Word* does not allow inserting a cross-reference without using the caption label. This creates a problem when referencing more than one item (in this case, tables) at the same time, like we do at the start of this paragraph. In order to solve this problem, we need to manually edit the inserted cross-reference field as follows. Insert the cross-reference as normal (select *References* ribbon; click on *Cross-reference*; select *Table* under *Reference type*; select *Only label and number*  under *Insert reference to*; select the table that you wish to refer to and then click *Insert*). This will create a cross-reference like Table 3.1. To remove the label and retain only the table number, click anywhere on this cross-reference and press SHIFT-F9 on your keyboard. This should reveal the field code used by *Word* to keep track of the cross-reference and should look something like { REF \_Ref119834227 \h } (the label will be different in your case). To display just the cross-reference number, insert the text \# "0.0" after \h in the field code to make it look like { REF \_Ref119834227 \h \# "0.0" }. With the cursor still somewhere in the field code press F9 on your keyboard. This should now display just the cross-reference number.

Table 3.1 Simple table example.

| Header 1 | Header 2 | Header 3 |
| --- | --- | --- |
| 1 | 2.3 | Orange |
| 2 | 100.5 | Blue |
| 3 | 35.0 | Black |

Notice that the table header is applied the style *Table Col Head*, whereas the body of the table is applied the *Table Body* style. Sometimes you may need to modify these styles to adapt to the data being displayed. For example, in Table 3.1 the data in the second column contains numbers accurate to the first decimal point. To centre these properly in the column with the numbers aligned based on the position of the decimal point, you need to insert a decimal tab at the centre of the column and align the column to be left justified.

Table 3.2 shows a more complicated table example. Notice that the column sub-headers are assigned the style *Table Col Subhead*. Also note that even though the numbers under the third subheading are without a decimal point, they are still aligned on their virtual decimal point position as explained in the previous paragraph.

Table 3.2 A more complicated table example.

| Table Head | Table Column Head | | |
| --- | --- | --- | --- |
| Table column subhead 1 | Table column subhead 2 | Table column subhead 3 |
| Item 1 | Item 2 | Item 3 | 100 |
| Item 4 | Item 5 | Item 6 | 2 |
| Item 7 | Item 8 | Item 9 | 10000 |

The paragraph immediately following a table should be assigned the *Dissertation Body after Table* style so as to maintain the correct spacing from the table.

### Lemmas, Theorems, Corollaries etc.

You can also use captions to number lemmas, theorems, corollaries and other items. To define a new numbering sequence, you need to define a new caption label in *Word*. Unfortunately, these new caption labels are not stored with the document itself, so you would need to define these yourself, on a need to basis.

For example, to define a new caption for lemmas, select the *References* ribbon and then click on *Insert Caption*. This will open a similar dialog as the one shown on the left of **Error! Reference source not found.** on page **Error! Bookmark not defined.**[[1]](#footnote-1). In the *Insert Caption* dialog box, create a new label by clicking the *New label …* button. Enter the label *Lemma* for this example (or any other item type, such as *Theorem* etc.) and press *OK*. Then click on the *Numbering …* button and in the resultant dialog box (similar to the one on the right-hand side of **Error! Reference source not found.**) tick the *Include chapter number* checkbox and select the period for the number separator. Once you define the new caption label, you can start using it in the same way as for inserting captions for figures.

The inserted caption will be applied automatically the *Caption* style. Instead, manually apply the *Labels* style to the newly inserted caption as shown in the example below for Lemma 3.1. To put the label (Lemma in this case) and the number in bold, select these manually and apply the style *Labels Bold*. When inserting a cross-reference in this case, the cross-reference will also be shown in bold within the text and you have to manually remove this character formatting.

Lemma 3.1 The Earth is a sphere. (Style: Labels)

Note that Lemmas 3.1 and 3.2 will be used to prove Theorem 3.1.

Lemma 3.2 The Earth rotates around the sun.

If a cross-reference becomes bold again on updating, then edit the corresponding field code to include the switches “\\* MERGEFORMAT” at the end of the field code.

Theorem 3.1 The Earth is one planet in the Solar System. (Style: Labels)

Proof: Using Lemmas 3.1 and 3.2 and observational data … (Style: Proof)

Note that the style for the Proof section is *Proof*, but you need to set the word *Proof* in italic by applying the style *Proof Italic* to it.

Captions for code listings may also be defined, as for Listing 3.1shown below.

Listing 3.1 Example code (Style: Labels)

# include <stdio.h> //(Style: Code Listing)

int main()

{

return 0;

}

Captions for algorithms may also be defined, as shown for Algorithm 3.1below. In the case of an algorithm, you need to convert the text to a table manually as sown in Algorithm 3.1. Note that the paragraph before an algorithm should have its *After spacing* set to 12 pt, as in this case. The Algorithm style used to write the algorithm proper automatically creates the line numbering.

|  |
| --- |
| Algorithm 3.1 Title of Algorithm (Style: Labels) |
| 1. inputs and outputs (Style: Algorithm) 2. **for** i: 1 … k **do** 4. … |

## Chapter Headers

In this template, automatic chapter headers appear on the top right-hand corner of each page. This enables easier navigation within the dissertation when reading it. In general, you do not need to change anything to make these work. However, if you insert new chapters, you must adhere to a couple of points not to break this feature.

When inserting a new *Heading 1* style, the style is automatically setup to start on a new page. However, this is not enough to make the chapter headings work correctly. You also need to insert a *Continuous Section Break* before the new chapter title. You can do this by moving the cursor at the start of the new chapter title. Then from the *Layout* ribbon click on *Breaks* and then click *Continuous* under the *Section Breaks*. The headers are set to be different in the first section page compared to the rest of the section so that the chapter header does not appear on the first page of the chapter. The headers are also linked to previous headers, so that you will not need to insert any field codes in the headers yourself. Notice that section breaks are in general hidden. To view these and other control characters in *Word* you need to toggle these by pressing CTRL-SHIFT-8 on the keyboard. You can also access this option from the *Home* ribbon by clicking the pilcrow button in the *Paragraph* section.

## Printing the Dissertation on Both Sides

This template is intended to create a dissertation that is printed only on one side of the paper (the left margin is wider than the right to allow for binding on the left). If you intend to print this dissertation on both sides of the paper, then you need to make some adjustments to facilitate this. It is highly recommended that you make a copy of the original document, just in case you may need to revert to the original single-sided version.

From the *Layout* ribbon, click on the corner arrow in the *Page Setup* section to open up the full range of page setup options and then select the *Layout* tab as shown in Figure 3.1. In the *Section start* drop-down box, select *Continuous*. In the *Headers and footers* section, tick the checkbox *Different odd and even*. In the *Apply to* drop-down box, select *Whole document*. Then click on the *OK* button. This correctly sets the gutter to be always on the inside of the page to enable correct binding.

You also need to correct the document’s headers and footers which are now different for odd and even numbered pages. Go to the *Abstract* page (page 2). You will notice that this now does not have page numbers. Double click in the footer area of this page, and you should enter the editor for the *Header and Footer* of the page. Over the footer area, you should see the label *Even Page Footer – Section 1*. From the *Header & Footer* ribbon, click on *Next* in the *Navigation* section. This should place you in the footer of the page labelled as *Odd Page Footer – Section 1*. Select the page number and copy and paste it in the previous *Even Page Footer – Section 1* by clicking on the *Previous* button to navigate back to the previous section.

Now navigate to the header area labelled *Even Page Header – Section 2*. If you retained the original document layout, this should land you on the second page of the Introduction chapter. In the *Header & Footer* ribbon, make sure that the *Link to Previous* selection is unselected.Then, from the *Insert* ribbon, click on the *Quick Parts* button in the *Text* section and then click on *Field*. In the dialog box that opens, select *StyleRef* under *Field names*. Under *Style names* select *Heading 1* and tick the check box next to *Insert paragraph number*. These selections are shown in Figure 3.2. Then press *OK*.

The above steps should insert the number 1 in the header (corresponding to chapter 1). Now, we need to insert the chapter title itself. Enter a space after the chapter number and then repeat the steps in the previous paragraph, but this time leave the *Insert paragraph number* checkbox unticked. This should insert the chapter title. The last step is to left-justify this header so that it always appears on the outside of the page.

Graphical user interface, application

Description automatically generated

Figure 3‑1 Converting document to printing on both sides.

You should also check the headers on other pages to make sure that everything appears correctly and to edit the headers as necessary.

Graphical user interface, application, Word

Description automatically generated

Figure 3‑2 Inserting chapter headers.

## Writing the Specification and Design Chapter

The purpose of this section is to give the reader a clear picture of the system/artifact/project/work that has been created in the FYP and why it has been created in the way chosen.

Details:

* Any design choices have to be justified (e.g., by discussing the implications of different design choices and then giving reasons for making the choices made).
* Fine details, specifically details of the system (software or hardware) should be left out. Also, any complete rigorous specification is better relegated to an appendix.
* Using diagrams (including but not limited to flowcharts and system level block diagrams) is strongly recommended.
* The design of the project will almost certainly have evolved during development. Focus should be made on the project as it is in its final state but often there are good reasons for describing intermediate states too (e.g., to discuss details of the design method used).

# Implementation

## Writing the Implementation Chapter

The Implementation section is similar to the Specification and Design section in that it describes the system but it does so at a finer level of detail, generally down to the code/theorem/algorithm/circuit/hardware… level. It can also describe any problems that may have arisen during implementation.

* In case of a software development describing of all the code in the system should be avoided as well as large “pieces” of code. Complete source code listings should be put on the accompanying digital media. In case of hardware the system should be divided into sub-systems or circuits that may be easily described and analysed.
* One must be especially critical to the operation of the system.
* Mentioning unforeseen problems encountered during implementation and how these are solved.

# Testing and/or Evaluation

## Writing the Evaluation and/or Evaluation Chapter

The testing and/or evaluation component of an FYP is critical.

* One has to make sure and explain why all tests used to evaluate the system are relevant, using evidence from the literature about similar systems, and justifying any deviations from standard approaches.
* Demonstration that system works as intended (or not, as the case may be).
* Include comprehensible summaries of the results of all critical tests that have been made.
* The student must also critically evaluate the system in the light of these tests results, describing its strengths and weaknesses.
* Ideas for improving it can be carried over into the Future Work section.
* Comparison of practical with theoretical results and their interpretation.
* Comparison with published work when available.

# Future Work

## Writing the Future Work Chapter

Whether by the end of the project all the original aims and objectives have been completed or not, there is always scope for future work. Also, the ideas will have evolved during the course of the project beyond the original target. The Future Work section is for expressing these ideas.

# Conclusions

## Writing the Conclusions Chapter

The Conclusions section should be a summary of the project and a restatement of its main results, i.e. what has been learnt and what it has achieved. An effective set of conclusions should not introduce new material. Instead, it should draw out, summarise, combine and reiterate the main points that have been made in the body of the report and present opinions based on them.

The Conclusions section marks the end of the report proper.

References

[1] [1] Università ta' Malta. *"Plagiarism and Collusion Guidelines for students, academics, and Faculties/Institutes/Centres/School."* <https://www.um.edu.mt/__data/assets/pdf_file/0007/436651/UniversityGuidelinesonPlagiarism.pdf.> (accessed Nov. 22, 2022).

[2] R. Cockrum, D. Clark and Z. Mylona, "Motivating engineering students to write technical papers," in *FIE'99 Frontiers in Education, 29th Annual Frontiers in Education Conference - Designing the Future of Science and Engineering Education,* San Juan, Puerto Rico, Nov. 10-13, 1999, pp. 13B5/11.

[3] IEEE Periodicals. *"IEEE Reference Guide."* IEEE Author Center. <https://ieeeauthorcenter.ieee.org/wp-content/uploads/IEEE-Reference-Guide.pdf> (accessed Nov. 16, 2022).

[4] University of Malta Library Outreach Department. *"Harvard Referencing Style Guidelines."* <www.um.edu.mt.> <https://www.um.edu.mt/__data/assets/pdf_file/0007/353662/Harvard_Guide.pdf> (accessed Nov. 16, 2022).

[5] Y. Fang, Y. Zhang and C. Huang, "CyberEyes: Cybersecurity Entity Recognition Model Based on Graph Convolutional Network," *The Computer Journal,* vol. 64, *(8),* pp. 1215–1225, 2020.

[6] C. Celia *et al*, "Self-directed learning and ICT," in *Innovation and ICT in Education: The Diversity of the 21st Century Classroom*, J. G. Galán, Ed. River Publishers, 2021, pp. 139-149.

stylefix

Bibliography

List here work that you have referred to during your work, but which you did not cite in the body of the dissertation. Same style and formatting rules as for references apply here (see Section **Error! Reference source not found.**). However, in the case of the IEEE style, there will be no citation numbers.

1. When to Use Appendices (Style: Appendix Heading 1)

When inserting a new appendix use the Appendix Heading 1 style. The appendix numbering and the word “Appendix” are inserted automatically.

* 1. Appendices Headings (Style: Appendix Heading 2)

When inserting appendix headings, use the corresponding Appendix Heading style according to the required level.

* + 1. Appendix Sub-Headings (Style: Appendix Heading 3)

Only three levels are defined for appendices.

* 1. What to put in Appendices

Appendices are repositories for material which the student wishes to include in the report but which would seriously obstruct the flow of ideas put anywhere in the main body. Copies of the final version of any code should be avoided – the code must be available digitally on accompanying media.

Examples of items that could go in appendices are:

* A glossary of terms.
* Fundamental and basic theory.
* Schematic Diagrams and PCB/IC layout snap shots.
* Detailed notes on the programming language chosen or hardware platform used or technology used in an IC environment.
* A user’s guide.

1. Version Information

|  |  |
| --- | --- |
| Version Number: | 1.1 |
| Date: | 30 November 2022 |
| Version Information: | * Changed fonts to Lato. * Given instructions how to install Lato font. * Clarified location of IEEE – Faculty of ICT style. |
| Author: | Prof Inġ Victor Buttigieg |
|  |  |
| Version Number: | 1.0 |
| Date: | 19 November 2022 |
| Version Information: | First version of the FYP dissertation template. |
| Author: | Prof Inġ Victor Buttigieg |

* 1. Copyright Information

The copyright of this template belongs to the Faculty of Information and Communication Technology (ICT), University of Malta. Students of the Faculty of ICT are allowed to use this template to write their FYP dissertation. No part of this template is allowed to be used by other third parties without prior authorisation.

* 1. Suggestions

If you have any suggestions how this template can be improved, or if you think that there is a specific issue with this template, please email these to [victor.buttigieg@um.edu.mt](mailto:victor.buttigieg@um.edu.mt).

1. When you are referencing a figure (or some other item) which is far away from the current position in the document, it is a good idea to include the page number where the item is located. You can do this using cross-referencing as well, by selecting *Page Number* under *Insert reference to* in the cross-reference dialog box. [↑](#footnote-ref-1)