

M.Sc. Thesis Checklist

It is not easy to write a good M.Sc.Eng. thesis, and it is perhaps even more difficult to come up with great general advice on how to do it. We have tried to summarize what should be in a technical report and some guidelines for how to write it in the following two documents:

<https://people.compute.dtu.dk/jerf/02609/writing.pdf>

<http://people.compute.dtu.dk/janba/MastersThesisAdvice.pdf>

Unfortunately, these documents cannot directly be used to check if you have done a good job. To make your lives easier, this document contains an actual checklist. Of course, being able to check off this list does **not** guarantee a high grade, but it may improve the odds. Probably, the list applies only to engineering students working in the field broadly defined as visual computing.

My thesis

- **is around 40-80 pages.** It can be shorter if you are brilliant and have done something truly novel. It should not be longer than 80 pages in general. If the thesis is very long there has to be a very good reason why all the bulk is needed.
- **contains at least 30 references.** At least 20 of these references are to either *scientific papers* that have appeared either in *academic journals* or been *presented at conferences, books* (popular books don't count), or *preprints*. The last category is a bit problematic since preprints have not been reviewed. Do not cite an arXiv paper unless you have good reason to believe it is solid. These days it is also impossible to avoid web pages. However, the bulk of your citations should **not** be to web pages or Wikipedia. To be clear, you should always start on Wikipedia or some very accessible source, but you should then find a more authoritative source and that is the one you cite. Finally, note that more is better when it comes to citations, but, of course, if a citation feels odd, you might be asked why you put it there.
- **contains many pictures of results.** In visual computing it is almost always possible to show results as pictures. We want many good pictures that clearly show what you have achieved.
- **contains several graphs and plots.** Graphs and plots are typically not results but say something about the quality of your results. It can be run-time as a function of input size, convergence plots for optimization or training, and plots can show how your work compares to some baseline. If there are no graphs and plots, you need to consider why you have no quantitative metrics.
- **contains some tables of numbers.** The tables probably show roughly the same information as the graphs but in a different form. In a table, it is often easier to see the combined statistics for something. For instance, in a table, we can easily show both run-time, memory usage, and some quality measure for several methods. That is hard to show in a plot because the Y axes are different.

- **has been carefully proofread for grammatical mistakes and typos.** This is becoming very easy to do, and there is just no excuse for handing in something unreadable. However, we want to read something written in your own voice - not that of an LLM.
- **contains several figures.** Figures are useful to illustrate principles and processes. It is rarely possible to explain something technical really well without using figures.
- **contains several equations.** While there are even academic papers that don't contain many equations, you need to consider if you know what you are doing if there is not a single equation in your thesis. In some theses there will be many equations on almost every page. That is not always required, though.
- **contains pseudocode or code snippets.** Often there is some algorithm that you worked on. We are honestly not going to read your code. Having key elements in the report is a good idea. Transforming it to pseudo code can be very helpful for readability, and if you can show the code as a (flow) diagram that is sometimes even better.

Some things are a bit harder to check but also very important. My thesis

- **explains the motivation for the project.** In the beginning of your thesis, you need to make it clear why anyone would ever care what you did. This has to be written in a very accessible way such that the reader does not have to read ahead in order to understand. The motivation does not have to explain precisely what you will do, but how your project might one day be a small part of saving the world.
- **adequately describes background.** Papers that you build directly upon are very important to cover, but you should also discuss more fundamental theory which is pertinent to your project. However, you have to draw a line. Undergraduate math, for instance, can certainly be considered known and does not have to be explained.
- **lays out your precise goals.** You need to tell the reader what you set out to do. Now your external examiner will have a copy of your original project plan. Probably, you did something slightly different. That is a good thing! You learned something and changed direction a bit. In other words, there is absolutely no shame in writing your goals with the benefit of hindsight. In effect, you tell us what your contributions are (what you have achieved) rather than your goal. However, it should be written in a forward looking way because it is in the beginning of your thesis - typically right after a background section.
- **is readable.** Often an external examiner will complain about the "red thread" (which may be a Danish or Scandinavian expression) or ask "who is the intended reader?". Usually, this means that he was reading something and had no idea why the information was needed or what it had to do with the work presented. Even worse, it could mean that you are explaining something in the report, and some piece of information that is crucial for understanding that part is missing. There is no shame in adding some text that guides the reader if you suspect that the thread is hard to follow.