

# IM520/MC505 Computer Vision

## Term Report

Peter A. Wiseguy

March 9, 2020

### **Abstract**

This document is adapted from the `HgbTermReport` template and based on the `hgbreport` LaTeX class, which is part of the `HagenbergThesis` document package. See <https://github.com/Digital-Media/HagenbergThesis> for the most recent version and additional materials (tutorial, manual etc.). Use this *Abstract* to provide a short summary of the contents in the remaining parts of the document. Note that it may be easier to place the individual chapters (“assignments”) in separate files and include them using `\include{..}`.

# Contents

# Guidelines for authoring lab reports

## Cumulative lab report

The lab report is a **single, cumulative document** which should contain a concise and well-structured summary of the work you did in this course. Also, you are asked to demonstrate and discuss your “report in progress” at any time throughout the semester. If help or advice is needed, please ask in class or use the course’s online forum.

## Weekly and final submissions

You are asked to upload a snapshot of your worked-out assignments weekly (i.e., prior to the next lab unit). These submissions are not graded but randomly checked to verify your progress. The final (complete) documentation for all assignments must be turned in at the end of the semester, prior to the (oral) exam. Thus you can pace your work individually and turn back to previous assignments for improvements at any later point.

**Note** that this **freedom** puts a lot of **responsibility** on yourself. Make sure that you start to write immediately, make steady progress and nothing important is left behind!

## Document structure and content

This document should help you to get started with the report. It is strongly suggested to use the final format right away to avoid surprises at a later point. Also, you will discover that writing and documenting your findings can help you in developing good and understandable solutions from the very beginning. Here are a few hints for writing your reports:

- One **chapter** should be dedicated to each **assignment** (note that chapter names have been modified for this).
- Make notes and write down your concepts immediately, that is, **before** you start coding!
- Describe each given task in your own words (do not just replicate the assignment). Then describe your approach, explain the main difficulties, clearly outline your solution, finally provide illustrative and meaningful results.
- Try to go beyond the material you find elsewhere, use and extend formal (mathematical) descriptions in a creative way. Also, try to keep your notation simple and

consistent, which is not always easy to do. Look at good examples and consider this part of the learning process.

- Be careful and creative when it comes to designing meaningful tests and selecting examples. Do not make screenshots but save the relevant images with ImageJ (usually as PNGs).
- Always give appropriate references to literature, figures and other work you used!
- Get used to work with formal and concise descriptions (math, symbols, relations, algorithms, ...) and train yourself in “getting the notation right”.
- Write in complete sentences and try to use a “professional” language.

## The bad and the ugly

- **Don’t just show program code!** Use prose with mathematical and algorithmic notation wherever appropriate (use the assignments and lecture notes for guidance). Insert actual code sparingly and only to show particularly interesting or critical parts of your implementation.
- **Do not explain details that are trivial** or elementary (such as Pythagoras’ law, for example). Otherwise, make a reference to the *all* sources you used (including school books, blogs, Wikipedia etc.).
- **Do not just replicate** equations and figures from the lecture materials, but – as said above – describe the task in your own words. In particular, you will be **executed** (i.e., beheaded, drawn and quartered) if you ever copy/paste equations from the assignment or any other sources. Make sure you write these things yourself (that’s what LaTeX is famous for)!

# Assignment 1

## Circle detection in binary dot images

The objective of the first laboratory is to find circle which is located inside black-white image and surrounded by additional, randomly distributed noise.

### 1.1 Introduction

While the definition of a problem might not seem to be that important at first, according to literature it is common problem.

## References