Digital 3D Geometry Processing - Project

Raphal Steinmann
Thomas Batschelet
Alain Milliet
Department of Computer Science, EPFL, Switzerland

***Abstract*—TODO**

I. Introduction

TODO

Abstract
: Short description of the whole paper, to help the reader decide whether to read it.

Introduction
: Describe your problem and state your contributions.

Models and Methods
: Describe your idea and how it was implemented to solve the problem. Survey the related work, giving credit where credit is due.

Results
: Show evidence to support your claims made in the introduction.

Discussion
: Discuss the strengths and weaknesses of your approach, based on the results. Point out the implications of your novel idea on the application concerned.

Summary
: Summarize your contributions in light of the new results.

II. Tips for Good Writing

A. Abstract

1) Example

B. Figures and Tables

Use examples and illustrations to clarify ideas and results. For example, by comparing Figure 1 and Figure 2, we can see the two different situations where Fourier and wavelet basis perform well.

C. Models and Methods

The models and methods section should describe what was done to answer the research question, describe how it was done, justify the experimental design, and explain how the results were analyzed.

The model refers to the underlying mathematical model or structure which you use to describe your problem, or that your solution is based on. The methods on the other hand, are the algorithms used to solve the problem. In some cases, the suggested method directly solves the problem, without having it stated in terms of an underlying model. Generally though it is a better practice to have the model figured out and stated clearly, rather than presenting a method without specifying the model. In this case, the method can be more easily evaluated in the task of fitting the given data to the underlying model.

The methods part of this section, is not a step-by-step, directive, protocol as you might see in your lab manual, but detailed enough such that an interested reader can reproduce your work [**?**], [**?**].

The methods section of a research paper provides the information by which a study's validity is judged. Therefore, it requires a clear and precise description of how an experiment was done, and the rationale for why specific experimental procedures were chosen. It is usually helpful to structure the methods section by [**?**]:

1) Layout the model you used to describe the problem or the solution.
2) Describing the algorithms used in the study, briefly including details such as hyperparameter values (e.g. thresholds), and preprocessing steps (e.g. normalizing the data to have mean value of zero).
3) Explaining how the materials were prepared, for example the images used and their resolution.
4) Describing the research protocol, for example which examples were used for estimating the parameters (training) and which were used for computing performance.
5) Explaining how measurements were made and what calculations were performed. Do not reproduce the full source code in the paper, but explain the key steps.

D. Results

Organize the results section based on the sequence of table and figures you include. Prepare the tables and figures as soon as all the data are analyzed and arrange them in the sequence that best presents your findings in a logical way. A good strategy is to note, on a draft of each table or figure, the one or two key results you want to address in the text portion of the results. The information from the figures is summarized in Table I.

Basis	Support	Suitable signals	Unsuitable signals
Fourier	global	sine like	localized
wavelet	local	localized	sine like

Table I
CHARACTERISTICS OF FOURIER AND WAVELET BASIS.

When reporting computational or measurement results, always report the mean (average value) along with a measure of variability (standard deviation(s) or standard error of the mean).

III. TIPS FOR GOOD SOFTWARE

There is a lot of literature (for example [**?**] and [**?**]) on how to write software. It is not the intention of this section to replace software engineering courses. However, in the interests of reproducible research [**?**], there are a few guidelines to make your reader happy:

- Have a `README` file that (at least) describes what your software does, and which commands to run to obtain results. Also mention anything special that needs to be set up, such as toolboxes[1].
- A list of authors and contributors can be included in a file called `AUTHORS`, acknowledging any help that you may have obtained. For small projects, this information is often also included in the `README`.
- Use meaningful filenames, and not `temp1.py`, `temp2.py`.
- Document your code. Each file should at least have a short description about its reason for existence. Non obvious steps in the code should be commented. Functions arguments and return values should be described.
- Describe how the results presented in your paper can be reproduced.

1) Equations: There are three types of equations available: inline equations, for example $y = mx+c$, which appear in the text, unnumbered equations

$$y = mx + c,$$

which are presented on a line on its own, and numbered equations

$$y = mx + c \tag{1}$$

which you can refer to at a later point (Equation (1)).

2) Tables and Figures: Tables and figures are "floating" objects, which means that the text can flow around it. Note that `figure*` and `table*` cause the corresponding figure or table to span both columns.

IV. SUMMARY

TODO

[1]For those who are particularly interested, other common structures can be found at http://en.wikipedia.org/wiki/README and http://www.gnu.org/software/womb/gnits/.