We Learned LaTeX So You Don't Have To: A Self-Documenting Template for Preparing Dissertations for the Rutgers Graduate School

BY Timothy Edmunds

DRAFT VERSION CREATED ON JANUARY 15, 2009

ABSTRACT OF THE DISSERTATION

We Learned LaTeX So You Don't Have To: A Self-Documenting Template for Preparing Dissertations for the Rutgers Graduate School

by Timothy Edmunds

Dissertation Director: Principal Advisor

The abstract of the dissertation.

The second paragraph of the abstract. ♠ **These paragraphs should obviously be replaced with a real abstract. (Notes like these are created with the** `\note` **command. They can be systematically hidden by setting the** `notes` **boolean in the document's root source file.)** ♠

Preface

The Graduate School Style Guide states:

> If a thesis or dissertation is composed in part or in full of whole chapters or independent articles or reports already published, the preface or acknowledgments page must indicate this and give citations to the earlier publications. Even if the portions of previous publications are more partial, such acknowledgement in the front matter is recommended.

Based on that requirement/recommendation, something like the following paragraph seems to suffice.

Portions of this dissertation are based on work previously published or submitted for publication by the author [Edmunds and Pai, 2006, 2008, 2009].

Summary of Changes

The preface also seems like a good place to put a summary of changes in iterative drafts (where change bars are enabled). Here is a list of (fake) changes that have been made to this template document.

- This Summary of Changes has been added to the preface.
- A section describing the document structure has been added to the introduction (Section 1.1).
- The caption in Figure 4.1 has been made even longer.
- The bibliography has been updated to reflect the acceptance for publication of a paper previously identified as under review [Edmunds and Pai, 2009].

Acknowledgements

The acknowledgements.

Dedication

The dedication.

Table of Contents

List of Tables

List of Figures

Chapter 1
Introduction

Pretty much every dissertation has an introduction. In fact, the Graduate School Style Guide expects that one will be present, and specifies that a formal heading must be used for the introduction.

This document is not actually a dissertation. It is both a template for a dissertation, and a guide to the use of the `ruthesis` document class and other packges/techniques to prepare dissertation manuscripts for the Rutgers Graduate School.

This document is formatted like a dissertation (since it is also the template for creating dissertation manuscripts). In Chapter 2, the features of the `ruthesis` document class are described. Chapter 3 describes how this template document is used with the `ruthesis` document class to prepare a dissertation manuscript. Some useful LaTeX techniques for dissertation preparation are shown in Chapter 4. Like any good dissertation, we draw conclusions in Chapter 5.

1.1 Document Structure

If the author is including endnotes at the end of each chapter (in accordance with one interpretation of the Graduate School's preferences), it might be a

good idea to let the reader know, by including something like the next paragraph at the end of the introduction.

For the convenience of the reader, a list of references is provided at the end of each chapter (where applicable). A bibliography containing all cited references is included at the end of the dissertation.

Chapter 2

`ruthesis` Document Class

The `ruthesis` document class (`ruthesis.cls`) is an extension of the `report` document class; it can be used to prepare dissertation manuscripts that comply with the style requirements of the Rutgers Graduate School.

2.1 Contributors

The `ruthesis` document class is the work of several contributors. The original `ruthesis.cls` was created by Les Clowney. It was based on the LaTeX document style file, `ruthesis.sty`, to which both Dave Steiner and Tara Madhyastha contributed. The document class was updated and combined with this dissertation template by Timothy Edmunds.

2.2 Document Options

The `ruthesis` document class defines the following options (in addition to the options provided by the report document class):

- `10pt`: typesets the manuscript with 10 pt font
- `11pt`: typesets the manuscript with 11 pt font
- `12pt`: typesets the manuscript with 12 pt font

- `electronic`: specifies that the manuscript include features suitable for electronic publication (but not unsuitable for hard-copy). (In practice, this option merely sets the `electronic` boolean, so that the author can condition such things as the use of the `hyperref` package using `\ifelectronic`.)

- `prelimdraft`: prepares a draft version of the manuscript (with a different title page, different headers and footers, etc.). Also sets the `prelimdraft` boolean so that the author can conditionally include material with `\ifprelimdraft`.

- `nonsubmission`: prepares a version of the manuscript that omits final-submission features (such as the signature block). Also sets the `nonsubmission` boolean so that the author can conditionally include material with `\ifnonsubmission`.

2.3 Booleans

The `ruthesis` document class defines the following booleans on which the manuscript author can condition using the `\if`*boolean*`\else\fi` LaTeX construct:

- `electronic`: set by the `electronic` document class option. true if the manuscript is to be prepared in a method suitable for electronic distribution (e.g., uses the `hyperref` package).

- `prelimdraft`: set by the `prelimdraft` document class option. true if the manuscript is to be prepared as a preliminary draft.

- `nonsubmission`: set by the `nonsubmission` document class option. true if the manuscript is not the final submission version (with signature block).

- `abstract`: set by the `\abstract` command. Determines whether the abstract is typeset. May be set to false by the user after the `\abstract` command in order to suppress the abstract.

- `preface`: set by the `\preface` command. Determines whether the preface is typeset. May be set to false by the user after the `\preface` command in order to suppress the preface.

- `acknowledgements`: set by the `\acknowledgements` command. Determines whether the acknowledgements page is typeset. May be set to false by the user after the `\acknowledgements` command in order to suppress the acknowledgements.

- `dedication`: set by the `\dedication` command. Determines whether the dedication is typeset. May be set to false by the user after the `\dedication` command in order to suppress the dedication.

- `abbreviationspage`: set by the `\abbreviationspage` command. Determines whether the List of Abbreviations is typeset. May be set to false by the user after the `\abbreviationspage` command in order to suppress the List of Abbreviations.

2.4 Switch Commands

Some of the commands provided by the ruthesis document class are one-way switches:

- `\copyrightpage`: includes a copyright page when typesetting the manuscript.

- \figurespage: includes a List of Figures when typesetting the manuscript.
- \tablespage: includes a List of Tables when typesetting the manuscript.
- \algorithmspage: includes a List of Algorithms when typesetting the manuscript.
- \phd: sets the degree being sought to be a Ph.D.
- \jointumdnj: indicates that the degree is offered jointly with UMDNJ.

2.5 Content Commands

The ruthesis document class defines the following commands that the author should use to specify certain types of content:

- \title{*The title of the dissertation*}
- \author{*The author of the dissertation*}
- \degree{*The degree being sought*} The default is Master of Science (or Doctor of Philosophy if \phd is used.
- \joint{*The institution offering the joint degree*}
- \director{*The principal advisor of the author*}
- \program{*The degree program*}
- \submissionyear{*Year of submission to the Graduate School*} The default is the current year.

- \submissionmonth{*Month of submission to the Graduate School*} The default is the current month.
- \approvals{*The number of committee member signature lines*}
- \abstract{*The abstract of the dissertation*}
- \acknowledgements{*The acknowledgements*}
- \dedication{*The dedication*}
- \preface{*The preface*}
- \abbreviationspage{*The abbreviations used in the dissertation*}

2.6 Environments

The new environments defined by the ruthesis document class are:

- `bibliographysection`: this environment should be used to produce the required single-spacing of the bibliography.
- `vita`: this environment should be used for the C.V. at the end of the manuscript.
- `achievementlist`: this environment can be used to create the employment history and publication list in the C.V.

2.7 Other Commands

Some other commands are defined by the ruthesis document class:

- `\beforepreface`: this command typesets all the material that comes before the preface (i.e., the title page, copyright page, abstract).
- `\afterpreface`: this command typesets all the material that comes between the `beforepreface` material and the body of the dissertation (i.e., the preface, the acknowledgements, the dedication, the table of contents, the lists of figures, tables and algorithms, etc.).
- `\appendix`: this command signals that all subsequent chapters are appendices.

2.8 Margins

The Graduate School Style Guide specifies that left margins must be 1.5 inches, and that the top, right, and bottom margins must be 1 inch. It is assumed that these margins apply to all pages in the manuscript. Since LaTeX is not always successful in complying with the specified margins (due to overfull boxes), the margins defined in the ruthesis document class are:

- Left: 108 pt (1.5″)
- Right: 92 pt ($1\frac{5}{18}$″)
- Top: 72 pt (1″)
- Bottom: 92 pt ($1\frac{5}{18}$″)

2.9 Single-Sided

The style guide never explicitly specifies whether the manuscript should be single- or double-sided, but the fact that the left margin is larger than the right

margin on all pages (not just odd-numbered pages) suggests that all pages are to be bound on the left. Therefore, the assumption is that pages should be single-sided — with electronic submission, this only really affects the hard-copy of the title page and abstract that need to handed in.

Chapter 3
Dissertation Template

As well as the `ruthesis` document class described in Chapter 2, this software bundle includes a dissertation template. When compiled, it generates this document, but the template is also intended to serve as an example of how to use the features of the `ruthesis` document class (and other LaTeX features) to construct a dissertation manuscript.

3.1 Contributors

The dissertation template was written by Timothy Edmunds.

3.2 Template Structure

Most conference or journal papers are compiled from a single LaTeX source file (along with off-the-shelf packages). However, given the large expected size of a dissertation, it is useful to make use of LaTeXś ability to break up the code for a large document into multiple source files; this is the approach that is taken with the dissertation template.

The root file of the template is `dissertation.tex`. That is the file that includes all the other source files that make up the template. It is also the file that is given as the argument to the compiler when building the document:

```
latex dissertation.tex
```

The root file is the head of a source tree that contains all the other content. The tree is a useful format for a dissertation source collection, since each chapter's source can be contained in its own directory (possibly including a sub-tree of sections, figures, or other supporting material).

The template's source tree includes a directory for the `header` (the document preamble), a directory for the `frontMatter` (abstract, preface, etc.), separate directories for each of the document chapters (`introduction`, `docClass`, `templateStructure`, `usefulTechniques`, `conclusion`), a directory for the `appendices`, and a directory for the `backMatter` (bibliography, C.V., etc.). Other directories in the source tree are the `bibliography` directory, which contains all the BIBTEX database files, and the `patches` directory, which contains bug-fixes for off-the-shelf packages. Of course, this tree structure can be rearranged to suit the needs of the dissertation (for example, BIBTEX database files could be placed within an individual chapter's directory if they are exclusive to that chapter).

3.2.1 `import` package

There are several ways to include other source files into the root source file in LATEX. The `\input{`*other file*`}` command inserts the external file during compilation as though it were included in whole in the original source file. The `\include{`*other file*`}` command does the same, except it typesets the included file on its own page(s). The `\include{`*other file*`}` command is equivalent to:

`\clearpage \input{`*other file*`} \clearpage`

Both of these methods suffer from the fact that the included file is interpreted as though it were part of the original source file; unless the original source file

is in the same directory as the included file, any relative links (such as further `\input` commands) will be garbled. This makes it difficult to use these commands to build a tree-like directory structure for the source files of a dissertation.

The `import` package addresses this situation by providing the

`\import{`*path*`}{`*file*`}`

command. This command preserves relative links (such as the search path for `\includegraphics`), so the material for each chapter can be organized into separate folders, without any need to refer to the overall directory structure within the chapter.

3.2.2 A Word About Editors

Dividing a dissertation's source into multiple files carries certain benefits (such as compartmentalization, making it easier to find code, simpler version control), but it can provide challenges for your editor. If you are using a very simple editor (i.e., one that edits text and doesn't do much else), then the challenges are few — you are already using the command line to compile your documents, and LaTeX itself is smart enough to handle all the `\input` or `\import` commands. However, if you are using an editor that is trying to be more of an integrated development environment, then that editor has to understand about multi-file documents.

Windows Editors

Use WinEdt. Seriously. Student licenses cost less than the publishing fee that the Graduate School is going to charge you when you submit your dissertation. Better yet, get your advisor to buy some site licenses for your lab; if your

advisor starts talking about how Emacs does everything you need to do, back away slowly, without making any sudden movements.

WinEdt does a good job of presenting the source code for a dissertation-sized LaTeX document to the user in a meaningful way, while simplifying the process of compiling the source into a pdf (or ps, or dvi). The IDE-like interface of WinEdt includes a side-bar navigation tree of the source code (deduced from WinEdt's understanding of `\input`, `\include`, and `\import` commands) and the document's labels and bibliography. The built-in compiler hooks (to the suggested LaTeX application, MikTeX) includes the ability to jump to errors in the source (to the best of LaTeX's admittedly limited ability to identify the source of an error).

Included with this template distribution is the WinEdt project file, `templateWinEdtProject.prj`. Using a project allows the author to specify which file is the root file of the document; whenever a compile command is executed, that file is used as the argument for the compiler, regardless of which file is currently being edited.

WinEdt also includes some syntax highlighting and other features that one would expect from an IDE.

OS X Editors

Unfortunately, there don't seem to be any editors with WinEdt's features and capabilities available on the OS X platform. TeXShop, an editor included in the MacTex distribution bundle, provides some of the IDE-like LaTeX editing features, but lacks WinEdt's project-type structure that allows for easy management of multi-file source trees. TeXShop needs to be told for each file in the source tree what the root document is, so that when the Typeset command is

issued, TeXShop doesn't just try to compile the current source file.

.*n.x Editors

Maybe you should go and ask your advisor about the wonders of Emacs.

3.3 Compiling

There are a number of different applications (even on the same platform) for compiling LaTeX source files into documents. When the end goal is an Adobe Portable Document Format (`.pdf`) file, the two common choices are:

- `pdflatex`: compile directly from the source to a `.pdf` file
- `latex`, `dvitops`, `pstopdf`: `latex` compiles the source into a `.dvi` file, `dvitops` converts the `.dvi` file to a `.ps` file, and `pstopdf` converts the `.ps` file to a `.pdf` file.

The output of these two different options is often similar, but not exactly the same. Also, depending on the method used, different source code is acceptable. For instance, when using `pdflatex`, supported graphics formats are `.pdf`, `.png`, and `.jpg`, but when using `latex`, `.pdf` graphics files are not allowed (but `.eps` files are) and bounding box information is not automatically extracted from raster image file formats (like `.png` and `.jpg`). Another difference is in support of line-wrapping links produced by the `hyperref` package. When compiled with the `pdftex` driver, `hyperref` links that would otherwise stick out into the margin (like this one) can be typeset on two lines (providing the link text can be broken, either naturally or according to `\-` hints); however, as noted in the `hyperref` README, the `dvips` driver does not support that

feature. The `hyperref` package's `breaklinks` option stops the text from being typeset in the margin, but the active link area is misplaced when the electronic document is viewed in a PDF viewer.

The usual course of action is to pick one of these two compiling schemes (`pdflatex` is recommended), stick with it, and not worry about code that doesn't compile with the other method. This option is particularly attractive for a dissertation, where the author does not have to worry about interoperability with a co-author's development environment. To maintain generality though, this template has been designed to work with either of the two compiler schemes mentioned above. To make this possible, some use has been made of the `ifpdf` package (which provides the `\ifpdf` test that signals whether the document is being compiled directly to PDF). Specifically, to deal with the graphics file format incompatibility mentioned above, a different graphics file extension list is used depending on the result of the `\ifpdf` test. When compiling directly to PDF, the command `\DeclareGraphicsExtensions{.png,.jpg,.pdf}` is used to signal that `\includegraphics{someImage}` commands should search for files named `someImage.png`, `someImage.jpg`, or `someImage.pdf`. When compiling to an intermediate file format, `\DeclareGraphicsExtensions{.eps,.png,.jpg}` is used. To allow different chapters to have different variations of these search orders, the commands are issued at the beginning of chapters that include graphics (e.g., in usefulTechniques/chapterHeader.tex and appendices/manySubfigsChapterHeader.tex).

3.4 Content Control

As described in Chapter 2, the `ruthesis` document class includes a number of options that can be used to control the typesetting of the manuscript. However,

there are any number of aspects of the document that the author may wish to control throughout the development process. For this reason, it is useful to define author-set switches in a central location to control the typesetting of the dissertation. For example, if a committee member mentions that he doesn't like references at the end of each chapter, it is handy to be able to change a single switch from true to false, rather than having to go digging through multiple files to comment out all the source code that generates per-chapter bibliographies.

This template defines a number of such switches at the beginning of the root source file (in the preamble). The purpose of each switch is described in the adjoining comments, and some of the switches are mentioned specifically at various places in this document.

One set of switch is used specifically to control the inclusion of content. Dissertations tend to be long documents, and it is often beneficial to be able to typeset just a portion (a chapter or two). To control the inclusion of content, this template defines a switch for each chapter (e.g., `\ifTemplateStructure` for this chapter); then whenever commands are issued that depend on the assumption that the chapter is to be typeset (like the `\import` command that includes the chapter), the commands are embedded in a check of the switch. For example:

```
\ifTemplateStructure%
  \chapter{Dissertation Template}%
  \label{sec:templateStructure}%
  \import{templateStructure/}{templateStructure}%
\else\fi% end ifTemplateStructure
```

Similar switches are used to control the inclusion of the C.V., the Acknowledgements, etc.

3.5 `natbib` for Better Bibliographies

The Graduate School has this to say about the format of References, Citations, and Bibliography:

> Footnotes at the bottom page[sic], endnotes at the ends of chapters or at the end of manuscript. Number notes consecutively. When notes are at the end of chapters, each chapter's notes should begin with the number one (1). Be consistent throughout and conform to generally accepted practice in the discipline.

There is a lot of room for self-contradiction in these instructions; for example, the generally accepted practice in your discipline may be to order references alphabetically, rather than by citation order. Also, some disciplines may use author-date style citations, rather than numbers. For this reason, this template distribution includes some different options for how to include bibliographies in the manuscript.

Of course, part of the appeal of preparing a manuscript in LaTeX is that references can be automatically selected for inclusion from a database and typeset according to a BibTeX style. However, basic BibTeX technology has been improved on by more powerful LaTeX packages. The specific package recommended (and supported by this template) is `natbib`. The `natbib` package is a reimplementation of LaTeX's `\cite` functionality that allows for more configurable citation styles (such as author-year, numerical, annotated). `natbib` also works with the `hyperref` package to create hyperlinks from citation to bibliography (and, with the `backref` option, from the bibliography to the citation). From an author's perspective, the main change that comes with using the `natbib` package is that use of the `\cite` command is replaced by either `\citep`

(for parenthetical citations — like this [Edmunds et al., 2005]) or `\citet` (for textual citations — like those used by Kaufman et al. [2005]). For more information on the citation capabilities of the `natbib` package, check out the Reference sheet for `natbib` usage.

3.5.1 BIBTEX Style Files

As mentioned above, the Graduate School Style Guide does not specify one set style for the typesetting of citations and bibliographies (except that bibliographies must be single-spaced). In some disciplines, numerical citations may be the "generally accepted practice," while in others, the author-date style might be the norm. Also, the information included in the bibliography entries might vary. The typesetting of the bibliographic entries is controlled by a BIBTEX style file (`.bst`). In order to be able to do things like author-year style citations, `natbib` requires special `.bst` files. Included in this bundle are two different reference styles: `ruthesis.bst` and `ruthesisciteorder.bst`.

`ruthesis.bst` is a style file for use with author-year style citations; it produces a list of bibliography entries in alphabetical order (by first author's last name). Since space constraints are not usually a problem in dissertation manuscripts, relatively little abbreviation is used in the typesetting of the bibliographic entries.

The second BIBTEX style file, `ruthesisciteorder.bst`, is a `natbib` friendly style that sorts the bibliography entries by the order of citation (to comply with the Graduate School's instruction that notes should be numbered consecutively).

As well as specifying the appropriate BIBTEX style, `natbib` has to be properly configured to select the desired citation/bibliography scheme. In this template, the choice between using numerical citations (and hence sorting by citation order) or author-year citations is controlled by the `numericalcitation` boolean defined in the root source file. By setting the value of this switch, the author can select from one of these two citation schemes. (See `header/header.tex` to find out how the switch's value affects the setup of `natbib`.)

Custom Styles

Not all dissertation authors may be satisfied with these two bibliography style choices. A particularly likely scenario is that an author may want to change how the bibliographic entries are typeset (for example, to list authors as Last Name, First Name). Unfortunately, BIBTEX style files are not very easy to hand-edit. The best solution for authors who need to change the bibliography style is probably to use the `custom-bib` package to generate a new bibliography (this is the method that was used to create the style files discussed above).

The `custom-bib` package provides an interactive process for creating a `.bst` file for the desired typesetting style. The result of the interactive process is a batch job (`.dbj` file) that can be processed by LATEX to produce the `.bst` file. For completeness, the batch job files that were used to produce `ruthesis.bst` and `ruthesisciteorder.bst` are also included in this distribution (the files are `ruthesis.dbj` and `ruthesisciteorder.dbj`, respectively).

Note that after the automated generation of the `.bst` files, some hand-editing was done (as noted in the comments at the beginning of the `.bst` files) to provide `hyperref`-compatible D.O.I. links.

3.6 Handling Multiple Bibliographies

In Section 3.5 we described how `natbib` can be used to create attractive citations and bibliographies while satisfying some of the Graduate School's style dictums. However, we did not address how to achieve the "references at the end of each chapter" that the style guide refers to as one option. Even when only a single bibliography at the end of the dissertation is produced, this template uses multiple bibliographies, since the C.V. publication list is produced by BibTeX (now would be a good time to mention that in addition to the two `.bst` files discussed in Section 3.5.1, this bundle also includes a BibTeX style file for chronological publication lists).

The default behaviour of LaTeX is to handle a single bibliography in a document, but there are a number of LaTeX packages that provide support for multiple bibliographies (e.g., `chapterbib`, `multibib`). The package used by this template is `bibunits`; this package has been found to work well with `natbib`, `hyperref`, and the template's structure. The `bibunits` package works by encapsulating portions of the document in a `bibunit` environment; within that `bibunit` environment, the `\putbib` command can be used to produce a bibliography containing the references cited within the `bibunit`. E.g.:

```
\begin{bibunit}%
%% Include the contents of a chapter
\import{}{someChapter}%
%% Create a bibliography section (defined by ruthesis.cls) at
%% the end of the chapter
\begin{bibliographysection}%
%% Specify each of the BibTeX database files that are
%% referenced in this chapter.
\putbib[bibliography/neuroControl,%
        bibliography/basketWeaving]%
\end{bibliographysection}%
\end{bibunit}%
```

In this template, the `bibunits` package is always used (since the C.V. publication list is produced as one `bibunit`), but the inclusion of a References section at the end of each chapter is controlled by the `perchapterbib` boolean defined at the beginning of the root source file.

The `bibunits` package also supports the generation of an overall bibliography that contains copies of the references from each of the `bibunits`. This feature is used to generate a bibliography at the end of the dissertation (if called for by the `endbib` boolean defined in the root source file). Note that the references in the C.V. publication list are excluded from this overall bibliography by calling the `bibunits` `\globalcitecopyfalse` command within the C.V. `bibunit`.

3.6.1 A Word About Editors

Many IDE-like LaTeX editors try to help automatically compile a document by executing a set pattern of commands (like `latex`, `bibtex`, `latex`, `latex`, `latex`), making certain filename assumptions (such as the assumption that `latex sourceName.tex` will produce a single auxiliary file, `sourceName.aux`, that contains all the document's citations, and that `bibtex` only needs to be executed on that one file). Beware of such assumptions; the `bibunits` package produces a separate `.aux` file for each `bibunit` (called `bu1.aux`, `bu2.aux`, `bu3.aux`, etc.). The good news is that up-to-date versions of the `bibtex` executable seem to handle `bibunits`'s output correctly. However, if the editor is trying to do its own processing to tell you what entries are in your bibliography, the editor needs to understand about `bibunits` as well.

Windows (WinEdt)

If you have an up-to-date version of WinEdt, then it knows what to do. To let WinEdt populate the list of references in your BIBTEX databases, include them all in a comment (one % only). E.g.:

```
%input "bibliography/neuroControl.bib"
%input "bibliography/basketWeaving.bib"
%input "bibliography/myPublications.bib"
```

OS X (TeXShop)

Recent versions of TeXShop include `latexmk`, a collection of perl scripts for executing `latex` (and auxiliary programs like `bibtex`) the requisite number of times to resolve cross references, etc. When the Typeset button is pressed, the command in the selection-list next to the button is executed; if `latexmk` is selected, all the necessary programs should be run (similarly with `pdflatexmk` for producing `.pdf`s directly)[1].

3.6.2 Troubleshooting

If the use of the `bibunits` package causes problems, and per-chapter bibliographies are not required, the use of the package can be eliminated, as long as the C.V. publication list is typeset manually. One way to do this is to create a separate `.tex` file with just one bibliography (the publication list), and copy the resulting `.bbl` file into the source for the C.V. Of course, that entire process

[1] Despite debugging efforts, using `latexmk` with TeXShop does not work with the PNG format image files used in this template. The conditional code that has been included to make `latex` happy with `.png` and `.jpg` files (see Section 3.3) works on the Windows `latex`→`dvi2ps`→`ps2pdf` pipeline, but `latexmk` produces improper `.dvi` files. Using `pdflatexmk` works.

has to be re-executed any time the content or formatting of the publication list needs to be changed.

References

Timothy Edmunds, S. Muthukrishnan, Subarna Sadhukhan, and Shinjiro Sueda. MoDB: Database system for synthesizing human motion. In *ICDE '05: Proceedings of the 21st International Conference on Data Engineering*, pages 1131–1132. IEEE Computer Society, Washington, DC, USA, 2005. ISBN 0-7695-2285-8. doi: 10.1109/ICDE.2005.89. 18

Danny M. Kaufman, Timothy Edmunds, and Dinesh K. Pai. Fast frictional dynamics for rigid bodies. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pages 946–956. ACM Press, New York, NY, USA, 2005. doi: 10.1145/1186822.1073295. 18

Chapter 4
Useful LaTeX Techniques for Dissertations

4.1 Introduction

In this chapter, we give examples of some LaTeX techniques that may be useful for producing pleasant dissertations. In order to help this template better display the capabilities of the ruthesis document class (and recommended packages), some meaningless material has been arbitrarily left in. This citation, for instance [Johansson, 1996].

Here is a paragraph that has (hypothetically) been changed since the last time it was sent to your advisor. If the `changebars` boolean is set in the document's root, this paragraph will be identified by a change-bar.

4.1.1 Related Work

The Related Work section is always a good place to define some important terms. *Important terms* are labels that you (or someone else) made up, possibly in an effort to elevate mundane concepts to a higher level of meaningfulness.

Of course, the Related Work [Robles-De-La-Torre and Hayward, 2001] section should also be rife with citations to the work of other researchers [Lloyd and Pai, 2001; Lederman and Klatzky, 2004]. Rife [Srinivasan et al., 1996].

In order to generate some nice cross-references in the bibliography(/ies), here are some more citations [Stein and Meredith, 1993; Calvert et al., 2004].

Here's a citation that you may recognize from earlier [Johansson, 1996].

Be sure to note that a nice feature of the `natbib` package is its support for both *parenthetical citations* (like the ones used so far) and *textual citations* (where the authors of the work are typeset into the main text. This is the kind of thing that Salcudean and Vlaar [1997] would probably have appreciated. Hwang et al. [2004] would have, too.

A respectable Related Work section should have a few [Hwang et al., 2004] more [Kuchenbecker et al., 2006; Okamura et al., 2001] citations [Lintern, 1991]. Definitely [Mertens, 1981; Wightman and Lintern, 1985].

4.1.2 Outline

This would be a good place to summarize the contributions of this chapter.

The remainder of this chapter is organized as follows. In Section 4.2 there's a nice demonstration of a row of subfigures. We then make a brief digression (due to page layout requirements) in Section 4.3 to show how small figures (or tables) can be nested within the body of the text. Returning to the world of subfigures, there is an illustration of some trickery that you can do with blank figures and subfigures to create seemingly overlapping figures in Section 4.4. Section 4.5 has some equations that use the `align` environment to line up nicely. Some tall and narrow figures are shown in Section 4.6. In Section 4.7 we draw conclusions.

4.2 Subfigures

Before we get to subfigures, let's start with an ordinary figure (see Figure 4.1). Most publications do not include a List of Figures; the inclusion of the List of

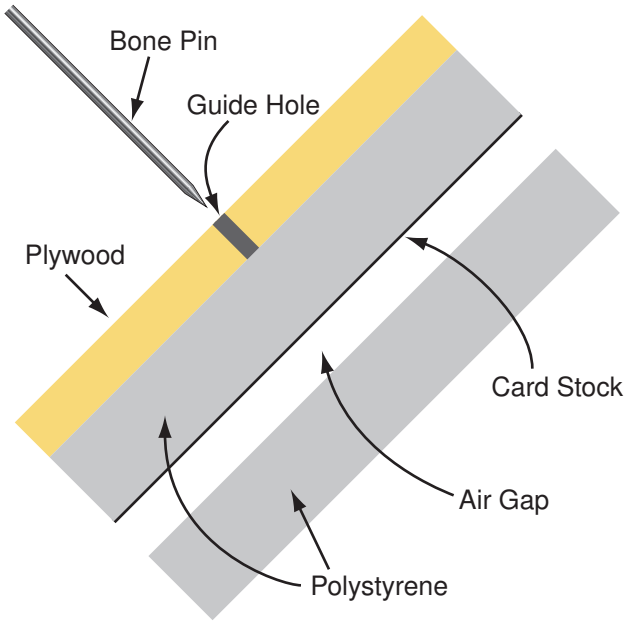
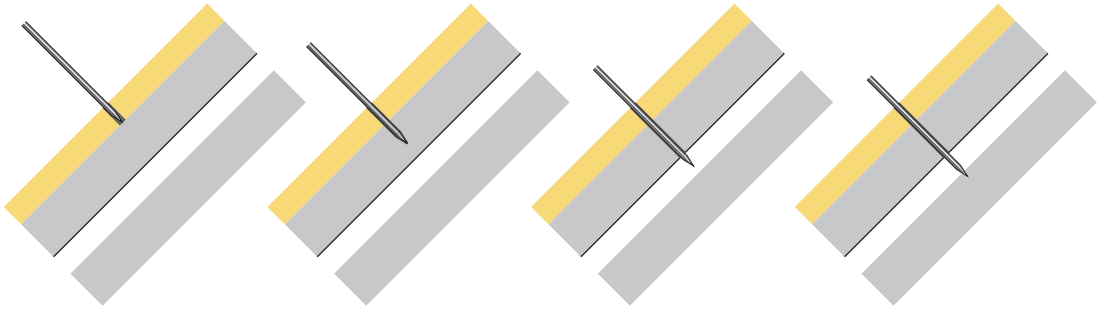


Figure 4.1: Experimental task setup (0.5× scale). A slab of polystyrene is laminated with card stock and separated from a second slab by an air gap. The pin's direction of movement is guided by a drill-hole through the covering plywood layer.

Figures (as required by the Graduate School Style Guide) requires attention to certain details. For example, the default behaviour is for the entire caption to be listed as the figure title in the List of Figures. However, since informative captions are almost invariably to long to be appropriate for the List of Figures, a short title can be specified as an optional argument to the `\caption` command; if specified, the short title will be used in the List of Figures. If the short title is specified but empty, the figure will not be included in the list of figures. Notice how Figure 4.1 has a long caption. However, if you look it up in the List of Figures, you will see that it has a tastefully short caption there.

So a basic figure is not hard. But when you want to illustrate a process over time (in this two-dimensional medium), you really need to be able to create a figure that has several subfigures arranged in a row. Like in Figure 4.2.



(a) Start (b) Carving (c) Complete (d) Failed

Figure 4.2: Task procedure. (a) The task starts with the tip of the pin resting on the surface of the polystyrene. (b) The subject must drive the sharpened pin through the first polystyrene slab. (c) The subject must stop before the pin touches the second slab. (d) If the pin penetrates the second slab, the task is failed.

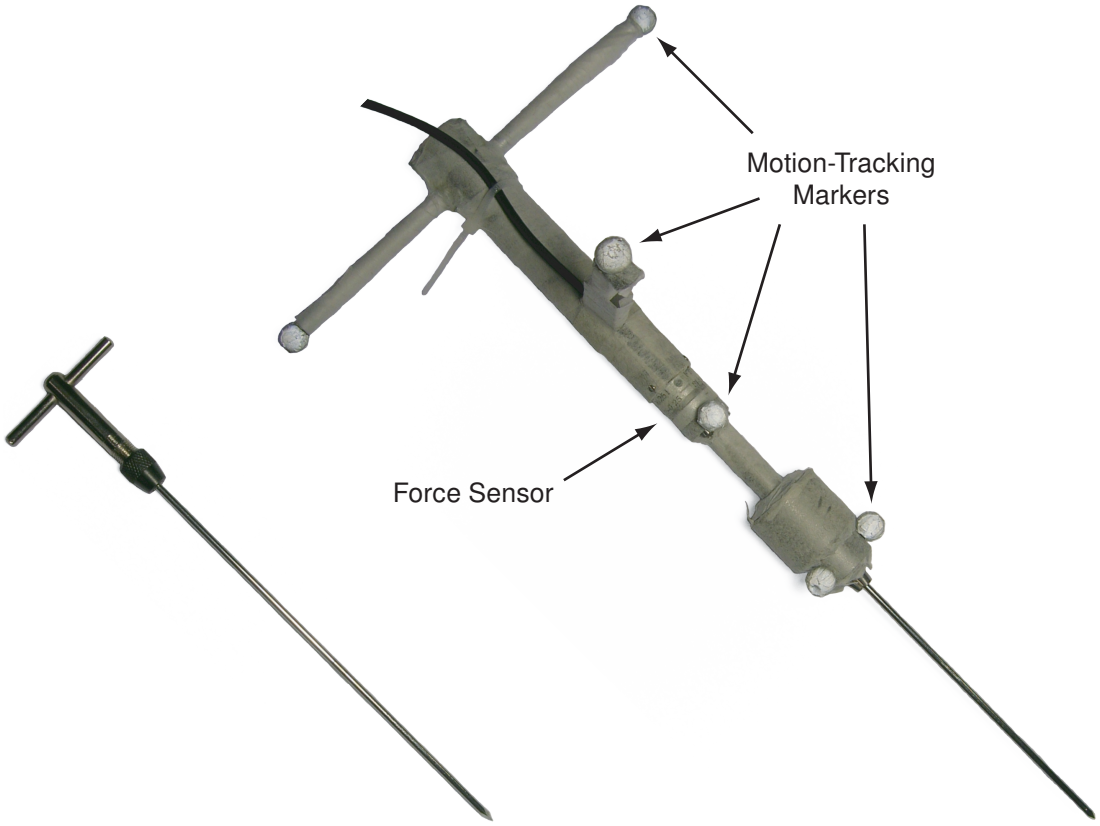
You may also want to refer to a subfigure within a figure (e.g., Figure 4.2b).

4.3 Wrapped Figures

Given the relatively large column-width of this single-column dissertation format, some figures (or tables) are simply too small to be placed in the entire margin-to-margin space that is the default. In this situation, one alternative is to find (or make) another small figure that fits well next to your problem child. If that solution is not tenable, then another alternative is to place the figure within the text in such a way that the surrounding text wraps fluidly around the figure. Table 4.1 is an example of this type of wrapped figure (or table). Wrapped figures and wrapped tables can be created using the

Target		Distractor	
μ_0	0.1	μ_0	0.1
a_1	0.783	a_1	0.25
a_2	0.1165	a_2	0.1
σ	0.05	σ	0.1

Table 4.1: Autoregression parameters used to generate the target and distractor textures.



(a) Bone Pin Holder (b) Instrumented Holder

Figure 4.3: Pin insertion tools (at $\sim$0.3$\times$ scale). (a) The bone pin is held in a T-handled pin vise. (b) To measure the force characteristics of the task, a pin holder was built that incorporated a force sensor and motion-tracking markers.

`floatflt` package.

4.4 More Subfigures

If all the figures you ever want to include are rectangular, read no further. Sometimes though, you have two figures that you want to present side-by-side, where allocating each one its own rectangle of space would be wasteful. If that is the case, an easy hack is to use some program (like Illustrator) to combine the two figures in a nicely overlapping way. But then how do you assign them subfigure labels? You cheat! Figure 4.3 is an example of this. It may look

like two different figures side-by-side...and it is! But the figure on the left is actually a blank image. We just messed around with the `viewport` argument to `\includegraphics` so that the subfigure captions are nicely positioned.

4.5 Aligned Equations

Here is an example of the use of the commands defined in the headers to make consistent typesetting easy. We defined a macro to make it easy to typeset this quantity, x_p, whether in math-mode or not. We also defined its derivative, $\dot{x_p}$.

The `amsmath` package provides the `align` environment that is nice for typesetting multi-row equations:

$$x_p = x_m \tag{4.1}$$

$$\mathbf{f}_m = 0 \tag{4.2}$$

The `align` environment can be used for more complicated equations too:

$$\mathbf{f}_m = \begin{cases} k(x_c - x_m), & x_m < x_c \\ 0, & x_m \geq x_c \end{cases} \tag{4.3}$$

$$x_p = \begin{cases} x_c, & x_m < x_c \\ x_m, & x_m \geq x_c \end{cases} \tag{4.4}$$

4.6 Tall Figures

For conference papers and journal papers, space constraints often dictate that your figures will have landscape aspect-ratios and sit at the top of a column of text. In this dissertation format though, it is often desirable to have a full-page portrait shaped figure. In this section, we give some examples of how to arrange this type of figure.

4.6.1 More Subfig Trickery

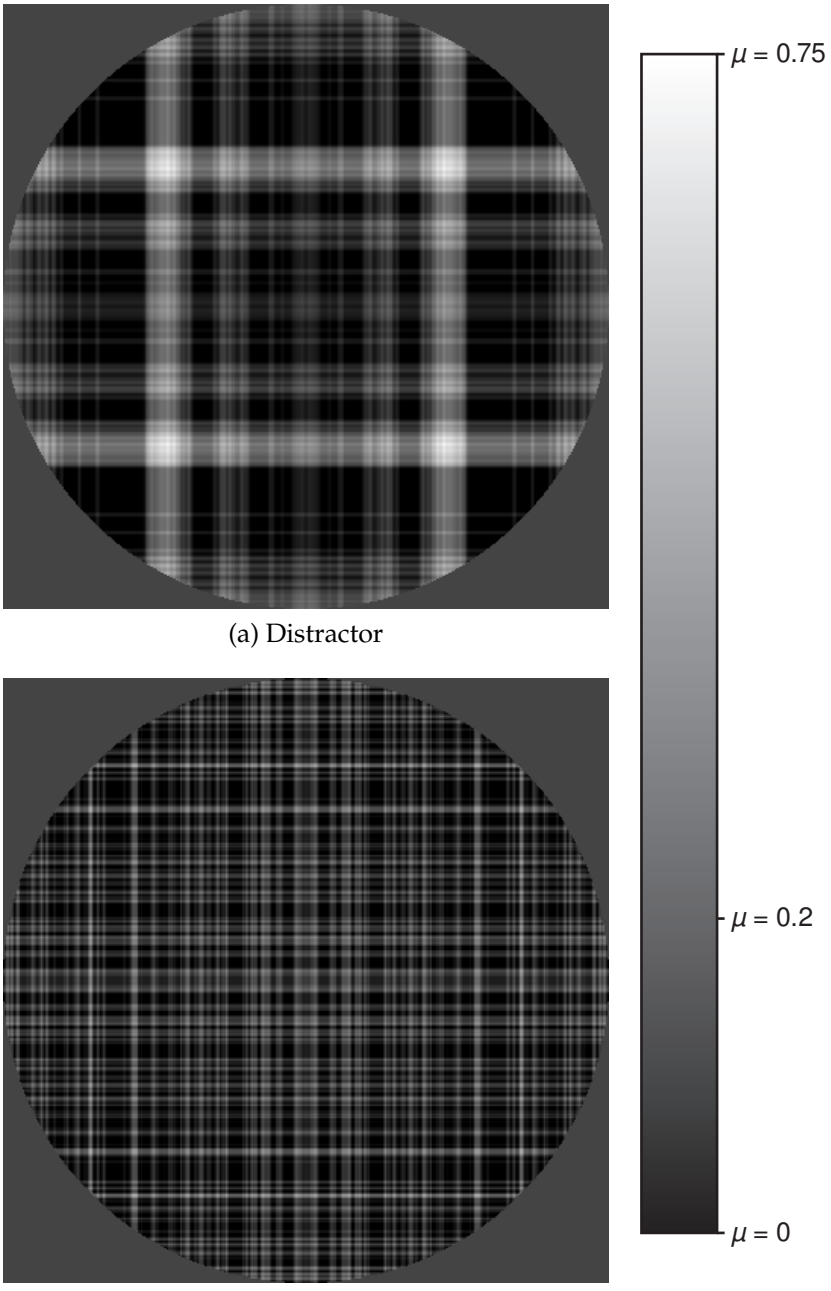
In Figure 4.4, `tabular` environments are used to arrange three subfigures in a full-page figure (a `figure*`). The caption for one of the subfigures is suppressed to make it seem like a legend.

4.6.2 Side Captions

It is sometimes desirable to have a figure's caption somewhere other than below it. This is particularly true for figures (or collections of subfigures) that are naturally tall and (relatively) narrow. See, for example, Figure 4.5. The `floatrow` package helps make that possible.

4.7 Conclusions

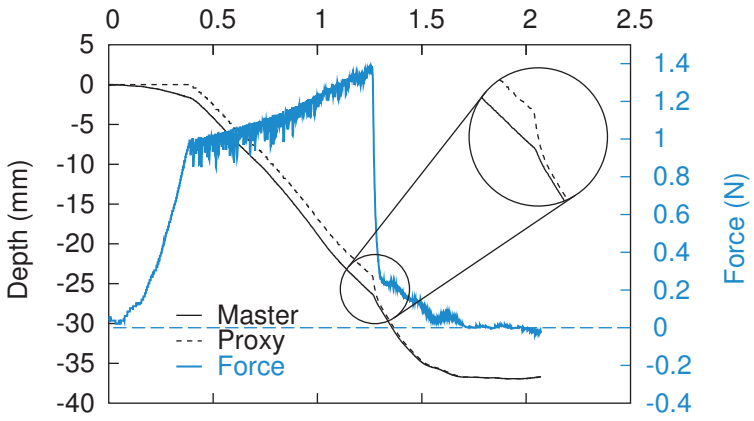
This chapter is over.



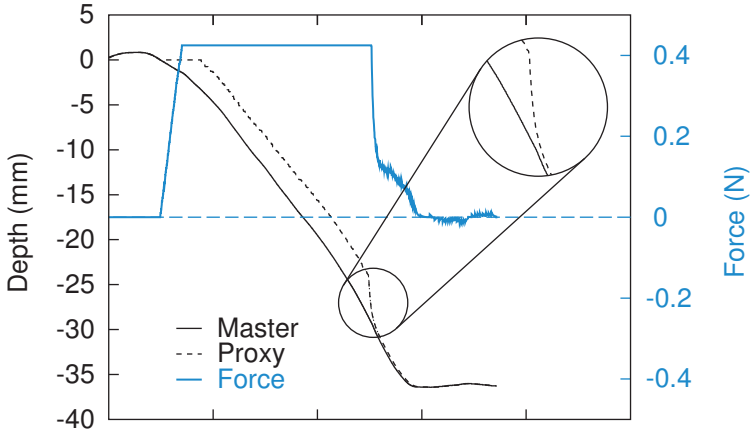
(a) Distractor

(b) Target

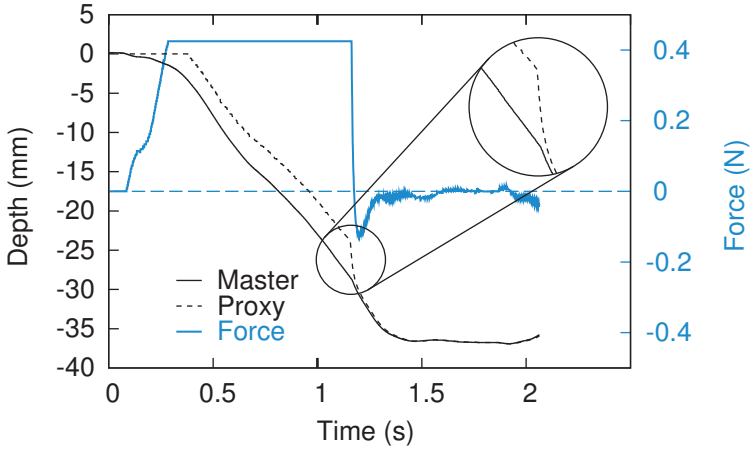
Figure 4.4: Example (a) distractor and (b) target texture patches at 2× scale. The texture is represented visually by lightness corresponding to the coefficient of friction, μ. The background surrounding both texture patches corresponds to the uniform coefficient of friction, $\mu_{background} = 0.2$. Note that although horizontal/vertical structure of the texture is readily apparent to the visual system, it is not perceived by haptic exploration.



(a) Full Stiffness



(b) Degraded Stiffness



(c) Augmented Low Stiffness

Figure 4.5:
The force/motion profiles for the different simulators.

(a) The full stiffness simulator reproduces both the high-frequency force discontinuities encountered during carving, and the sudden negative acceleration of the master upon emergence from the material.

(b) The degraded stiffness simulator saturates below the force levels at which high-frequency discontinuities occur and fails to generate significant master acceleration at the point of emergence.

(c) The open-loop force pulse applied in the augmented low stiffness simulator restores some of the master acceleration at the time of emergence from the material.

References

Gemma A. Calvert, Charles Spence, and Barry E. Stein, editors. *The Handbook of Multisensory Processes*. The MIT Press, 2004. 25, 34, 53

Jesse D. Hwang, Michael D. Williams, and Günter Niemeyer. Toward event-based haptics: Rendering contact using open-loop force pulses. In *HAPTICS '04: Proceedings of the 12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 24–31. 2004. ISBN 0-7695-2112-6. doi: 10.1109/HAPTIC.2004.1287174. 26

Roland. S. Johansson. Sensory control of dexterous manipulation in humans. In Alan M. Wing, Patrick Haggard, and J. Randall Flanagan, editors, *Hand and Brain: The Neurophysiology and Psychology of Hand Movements*, pages 381–414. Academic Press, 1996. 25, 26

Katherine J. Kuchenbecker, Jonathan Fiene, and Günter Niemeyer. Improving contact realism through event-based haptic feedback. *IEEE Transactions on Visualization and Computer Graphics*, 12(2):219–230, March–April 2006. ISSN 1077-2626. doi: 10.1109/TVCG.2006.32. 26

Susan J. Lederman and Roberta L. Klatzky. Multisensory texture perception. In Calvert et al. [2004], pages 107–122. 25

Gavan Lintern. An informational perspective on skill transfer in human-machine systems. *Human Factors*, 33(3):251–266, June 1991. ISSN 0018-7208. 26

John E. Lloyd and Dinesh K. Pai. Robotic mapping of friction and roughness for reality-based modeling. In *IEEE International Conference on Robotics and Automation*, pages 1884–1890. 2001. 25

Henry W. Mertens. Perception of runway image shape and approach angle magnitude by pilots in simulated night landing approaches. *Aviation, Space, and Environmental Medicine*, 52(7):373–386, July 1981. 26

Allison M. Okamura, Mark R. Cutkosky, and Jack Tigh Dennerlein. Reality-based models for vibration feedback in virtual environments. *IEEE/ASME Transactions on Mechatronics*, 6(3):245–252, September 2001. ISSN 1083-4435. doi: 10.1109/3516.951362. 26

Gabriel Robles-De-La-Torre and Vincent Hayward. Force can overcome object geometry in the perception of shape through active touch. *Nature*, 412:445–448, 2001. doi: 10.1038/35086588. 25

S. E. Salcudean and T. D. Vlaar. On the emulation of stiff walls and static friction with a magnetically levitated input/output device. *ASME Journal of Dynamic Systems, Measurement and Control*, 119(1):127–132, March 1997. ISSN 0022-0434. 26

M. A. Srinivasan, G. L. Beauregard, and D. L. Brock. The impact of visual information on the haptic perception of stiffness in virtual environments. In *ASME 1996: Proceedings of the ASME Dynamics Systems and Control Division*, volume 58, pages 555–559. ASME, 1996. 25

Barry E. Stein and M. Alex Meredith. *The Merging of the Senses*. Cognitive Neuroscience. The MIT Press, 1993. 25

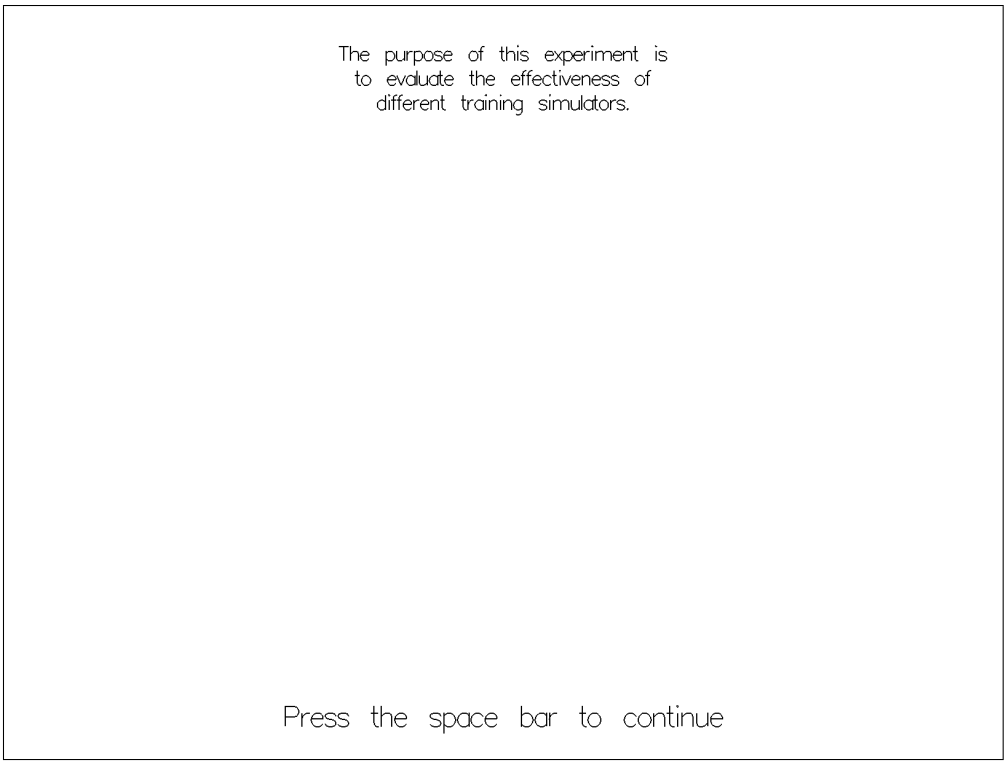
Dennis C. Wightman and Gavan Lintern. Part-task training for tracking and manual control. *Human Factors*, 27(3):267–283, June 1985. 26

Chapter 5
Conclusion

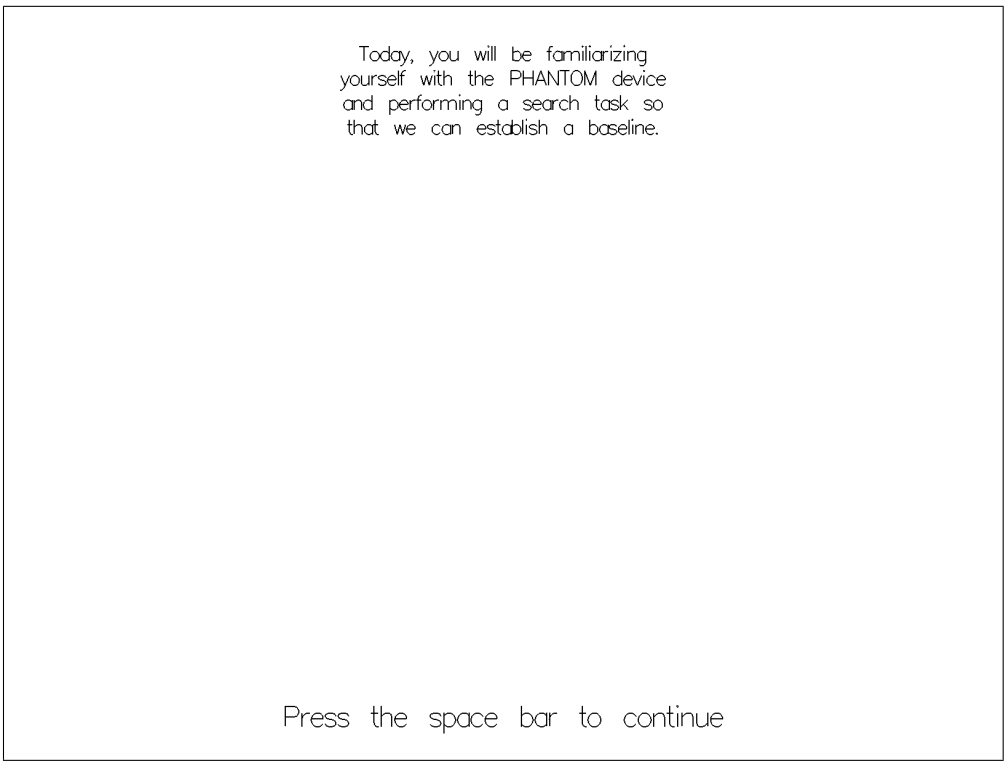
This dissertation is over.

Appendix A
Many Subfigures

This appendix illustrates the use of the `\ContinuedFloat` command (from the `subfig` package) to break a figure that includes several sub-figures (more than will fit on one page) into multiple figures. Also, the `alphalph` package is used to handle the creation of letter labels for situations where more than 26 entities are labelled. Figure A.1 is a figure with 28 subfigures. Note that Figure A.1aa is correctly labelled.

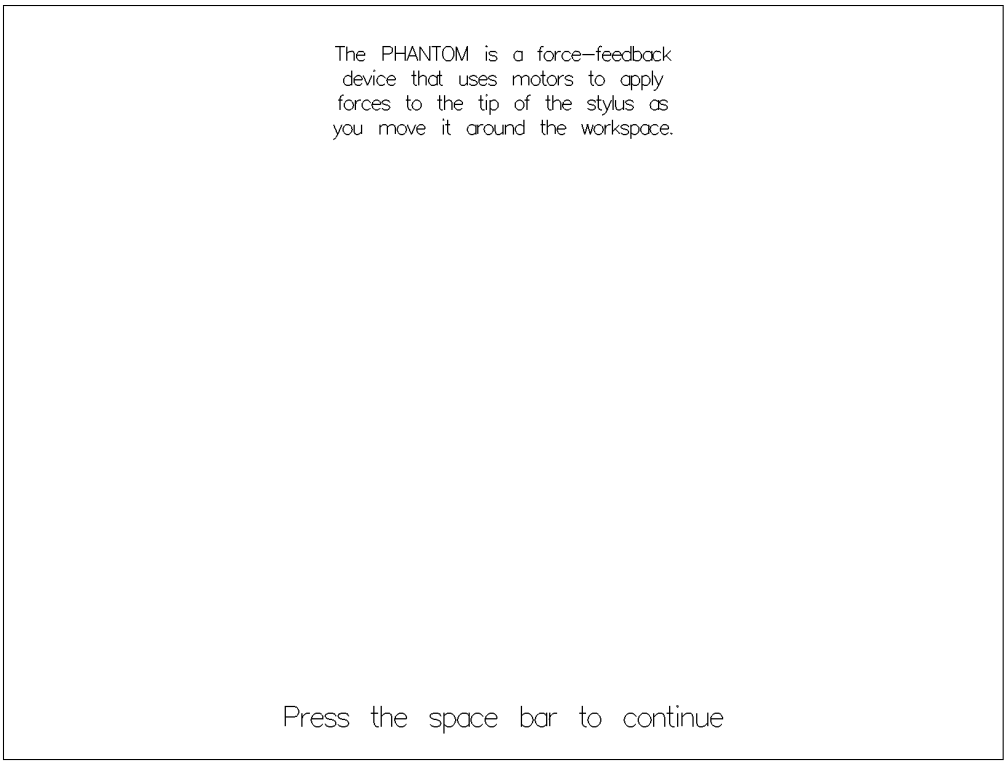


(a) The background/text colours have been inverted for clarity in this medium.

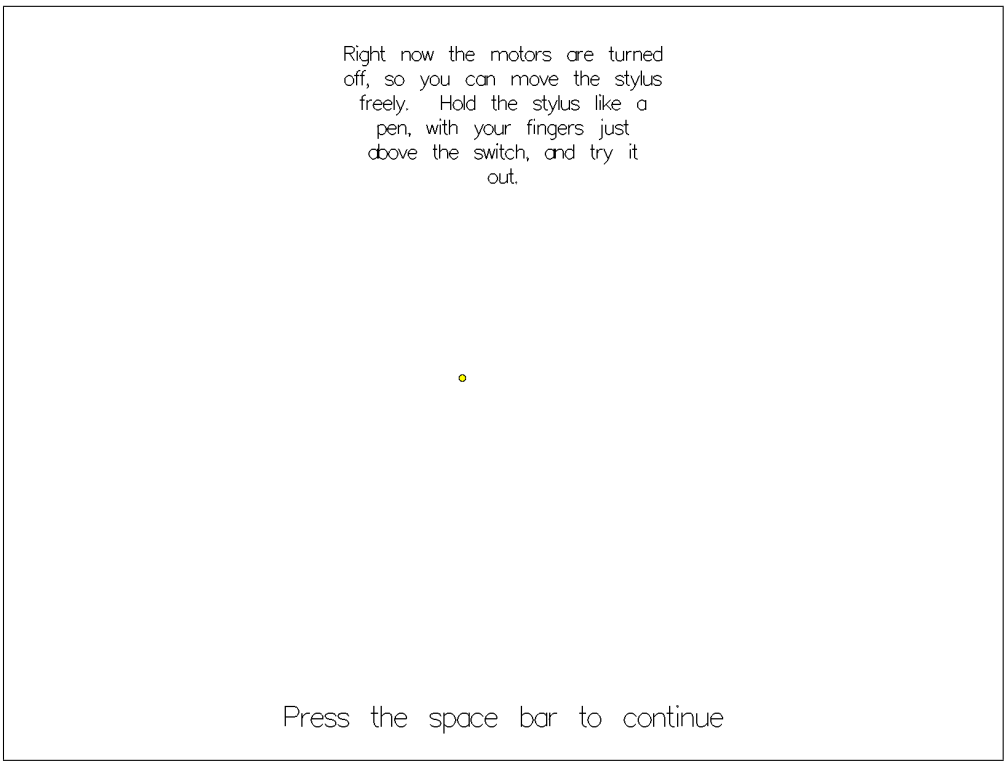


(b)

Figure A.1: Instructions presented to the subject (continued below).

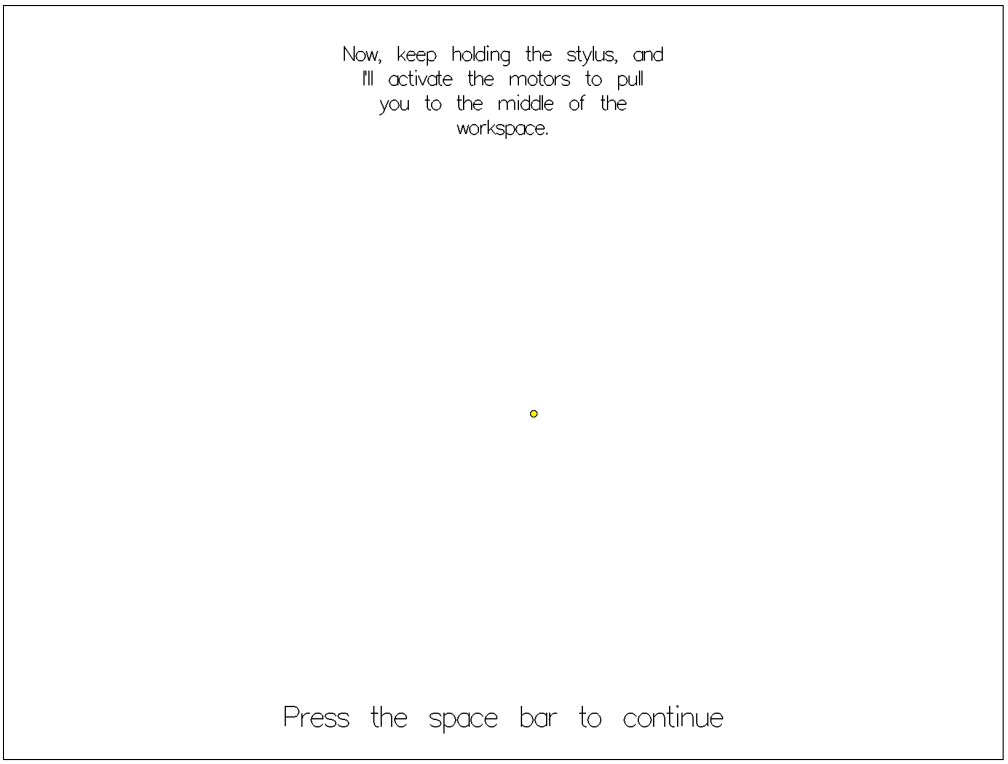


(c)

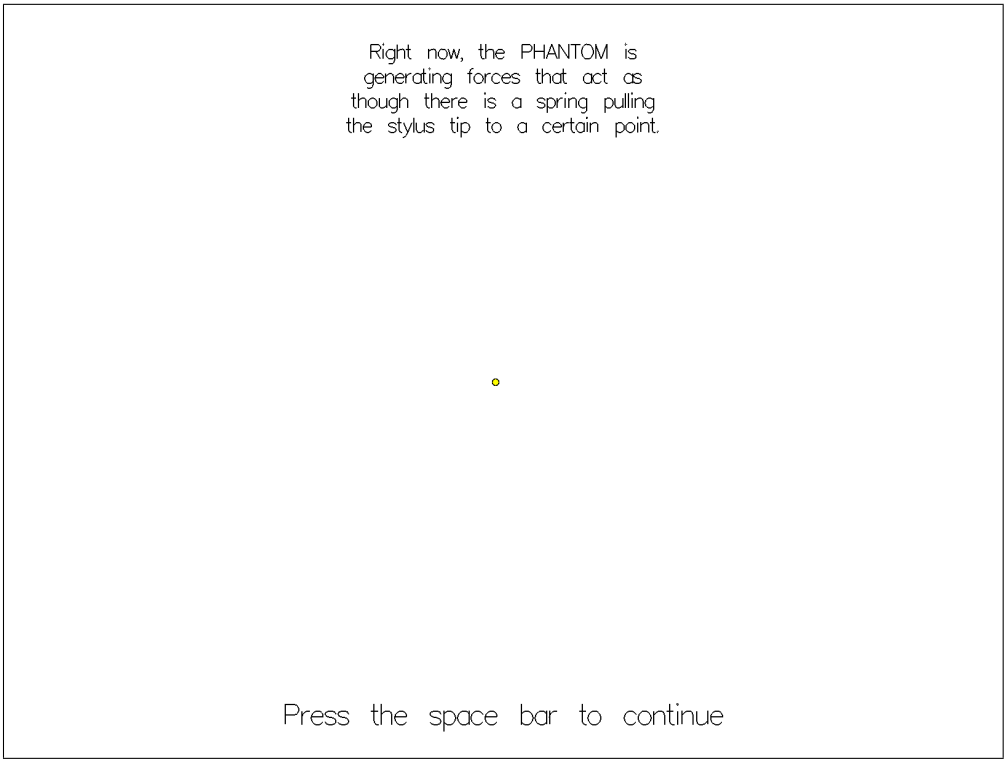


(d) The position of the haptic master within the workspace is indicated by the yellow sphere.

Figure A.1

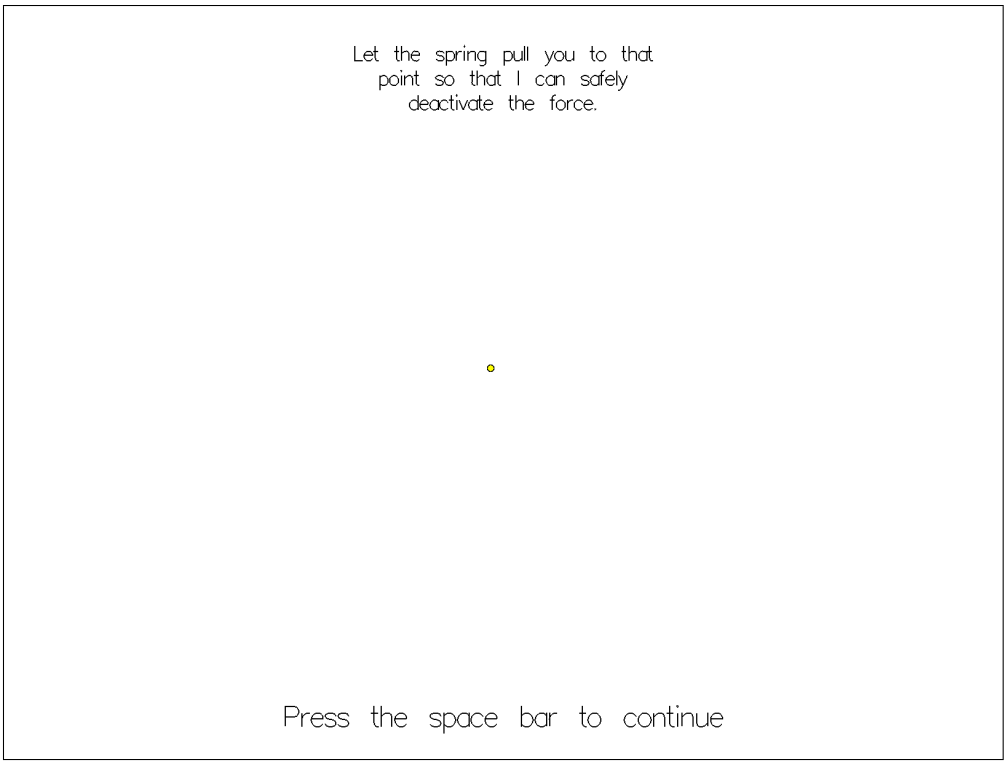

Now, keep holding the stylus, and
I'll activate the motors to pull
you to the middle of the
workspace.
Press the space bar to continue

(e)

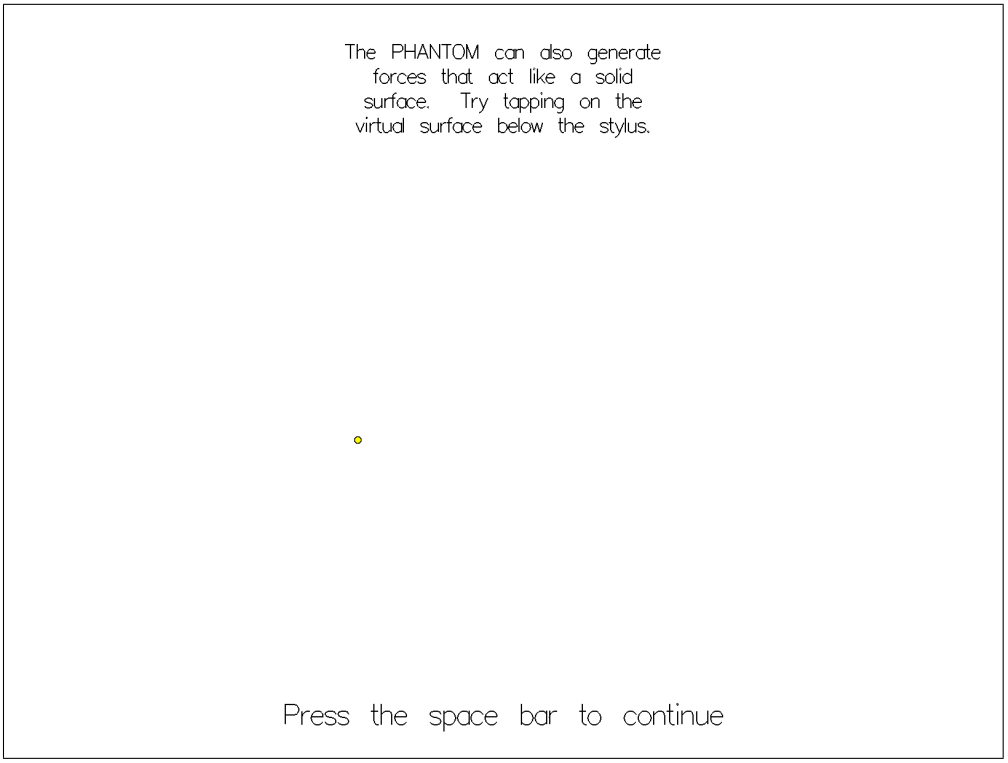

Right now, the PHANTOM is
generating forces that act as
though there is a spring pulling
the stylus tip to a certain point.
Press the space bar to continue

(f)

Figure A.1

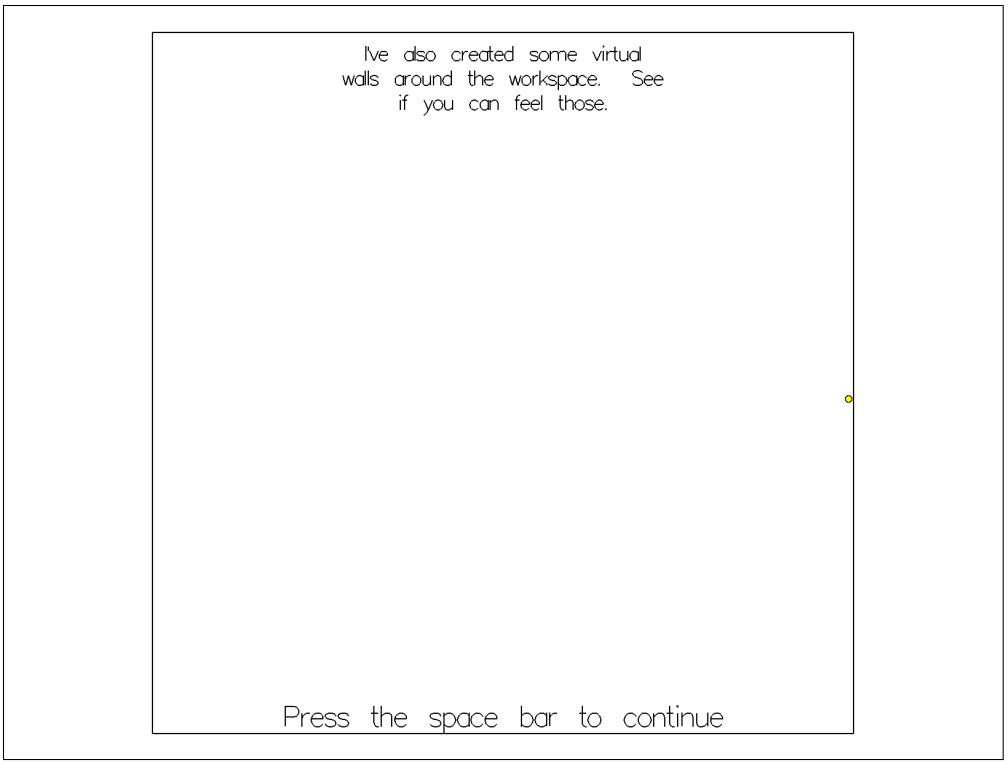

Let the spring pull you to that
point so that I can safely
deactivate the force.
Press the space bar to continue

(g)

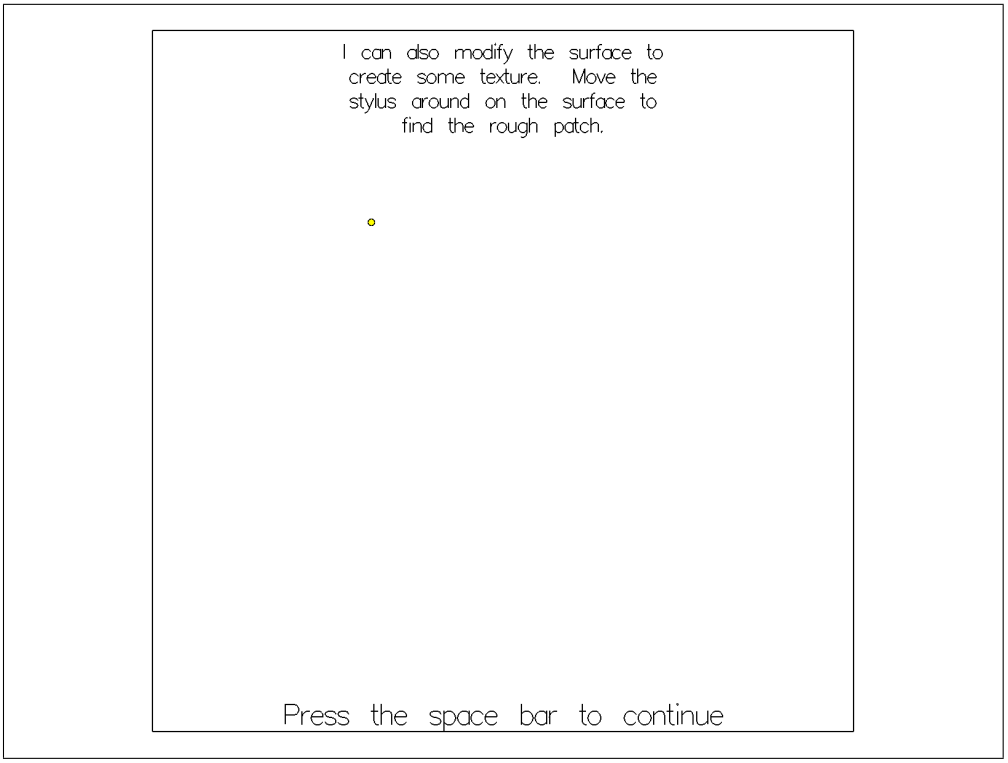

The PHANTOM can also generate
forces that act like a solid
surface. Try tapping on the
virtual surface below the stylus.
Press the space bar to continue

(h)

Figure A.1

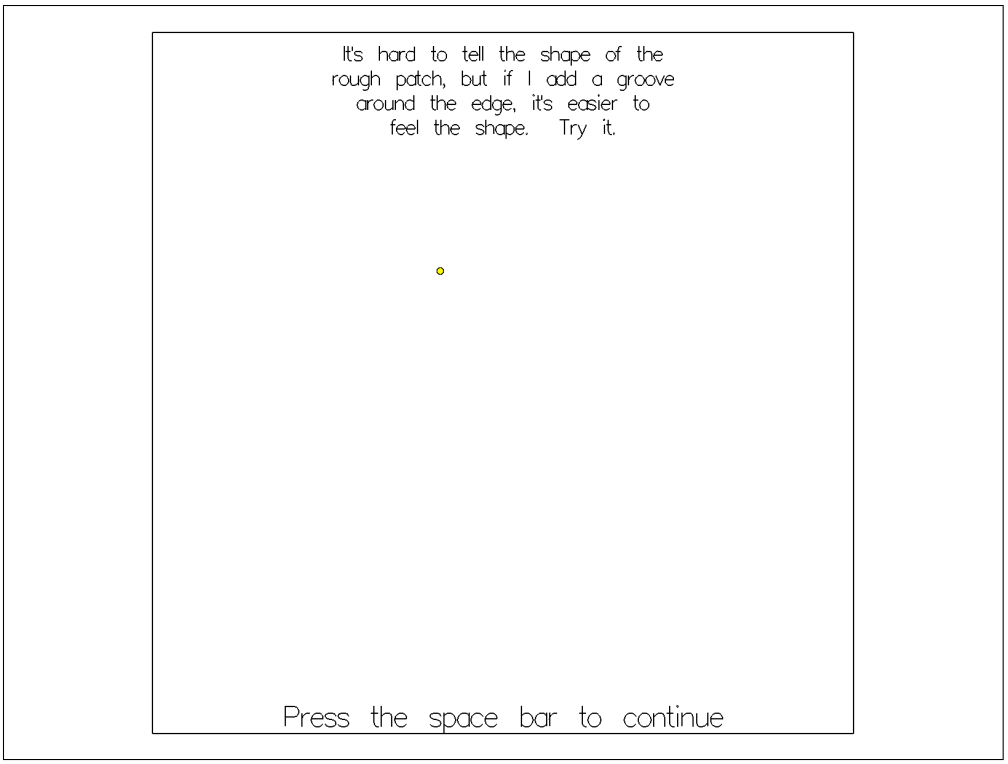

I've also created some virtual walls around the workspace. See if you can feel those.
Press the space bar to continue

(i)

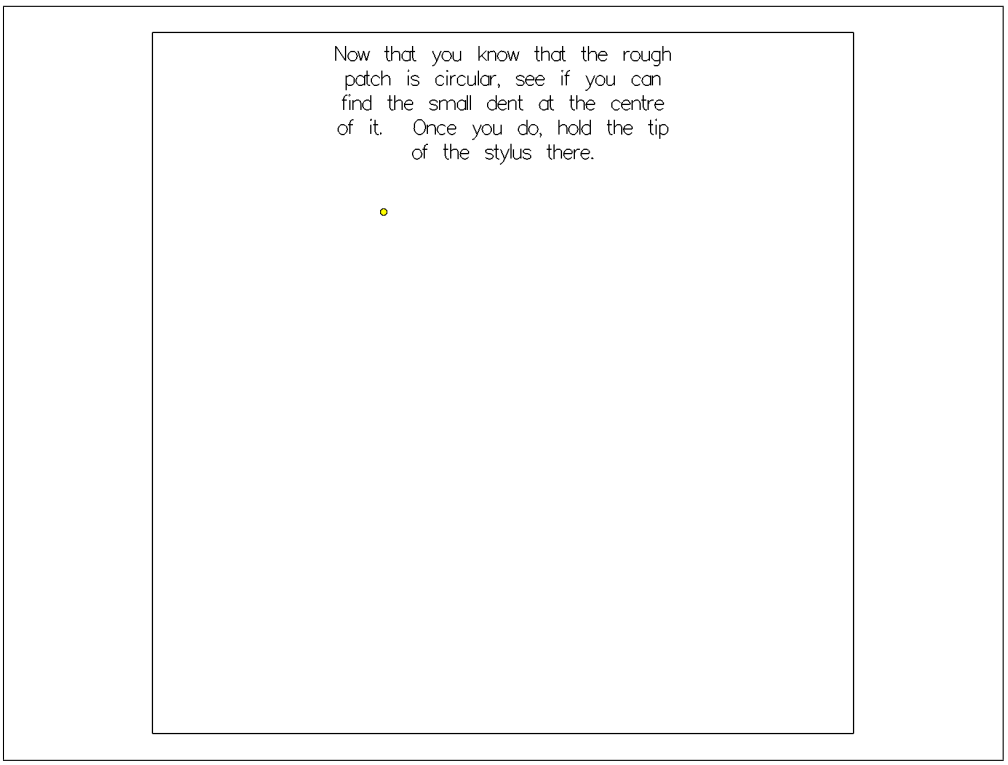

I can also modify the surface to create some texture. Move the stylus around on the surface to find the rough patch.
Press the space bar to continue

(j)

Figure A.1

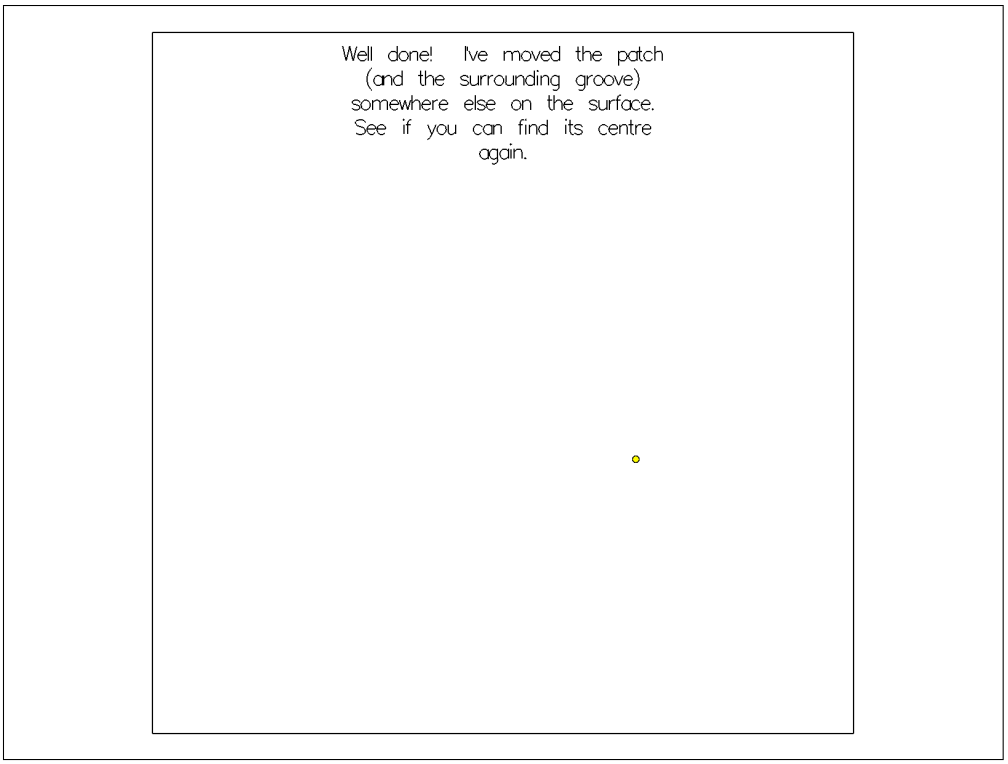

It's hard to tell the shape of the rough patch, but if I add a groove around the edge, it's easier to feel the shape. Try it.
Press the space bar to continue

(k)

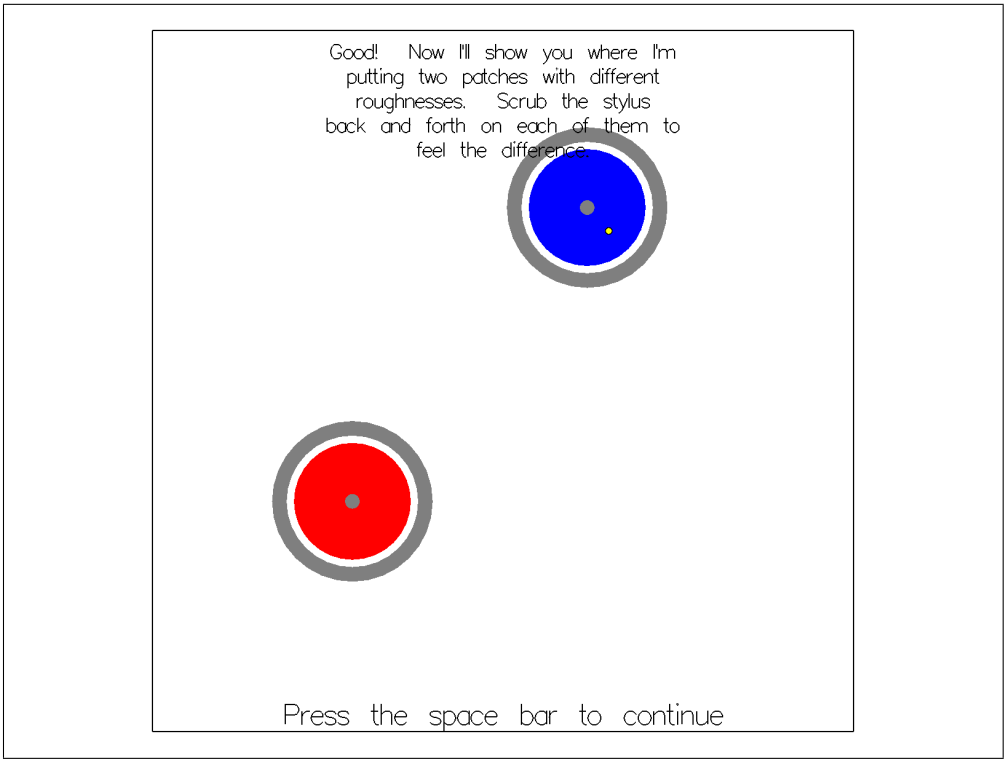

Now that you know that the rough patch is circular, see if you can find the small dent at the centre of it. Once you do, hold the tip of the stylus there.

(l)

Figure A.1

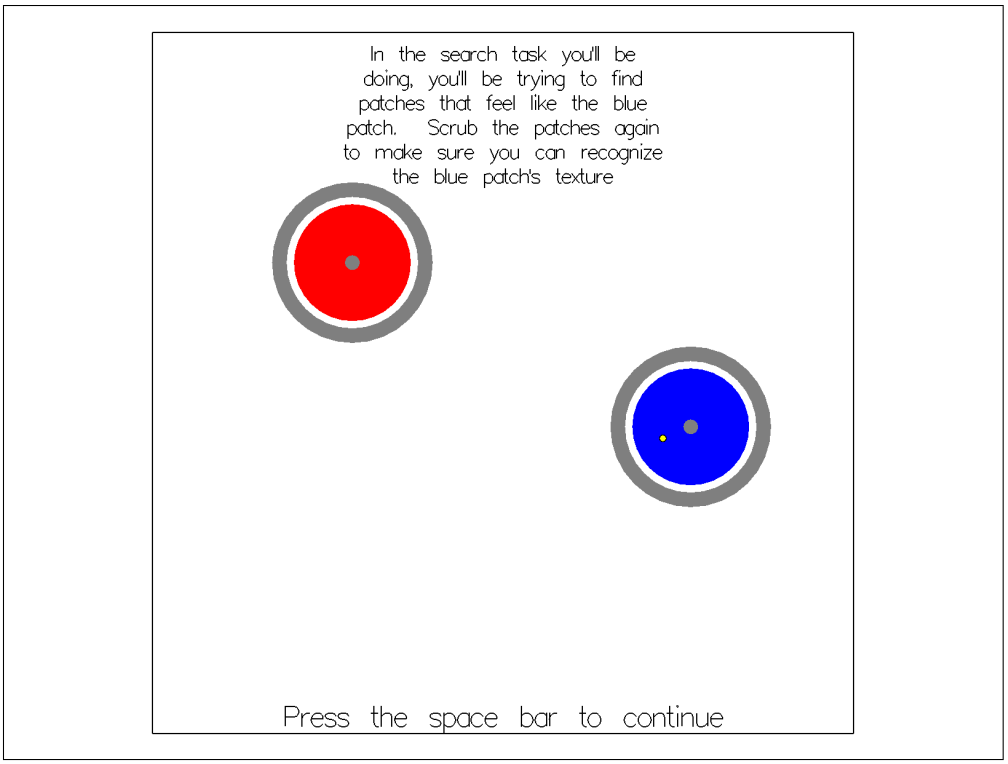

Well done! I've moved the patch
(and the surrounding groove)
somewhere else on the surface.
See if you can find its centre
again.

(m)

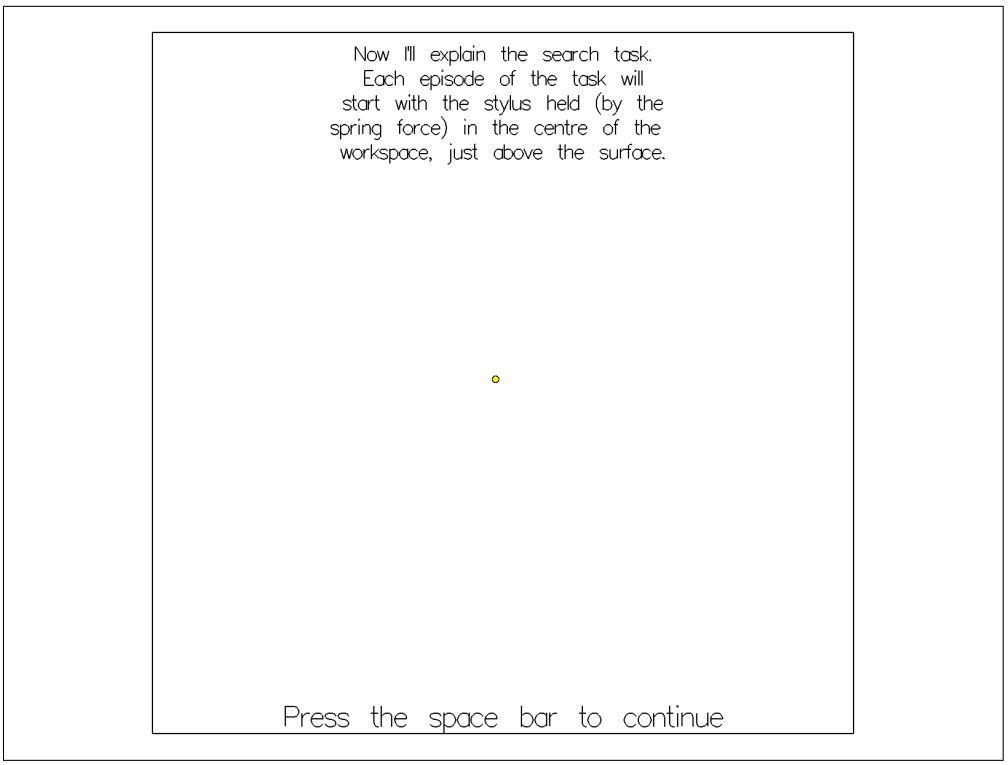

Good! Now I'll show you where I'm
putting two patches with different
roughnesses. Scrub the stylus
back and forth on each of them to
feel the difference.
Press the space bar to continue

(n)

Figure A.1

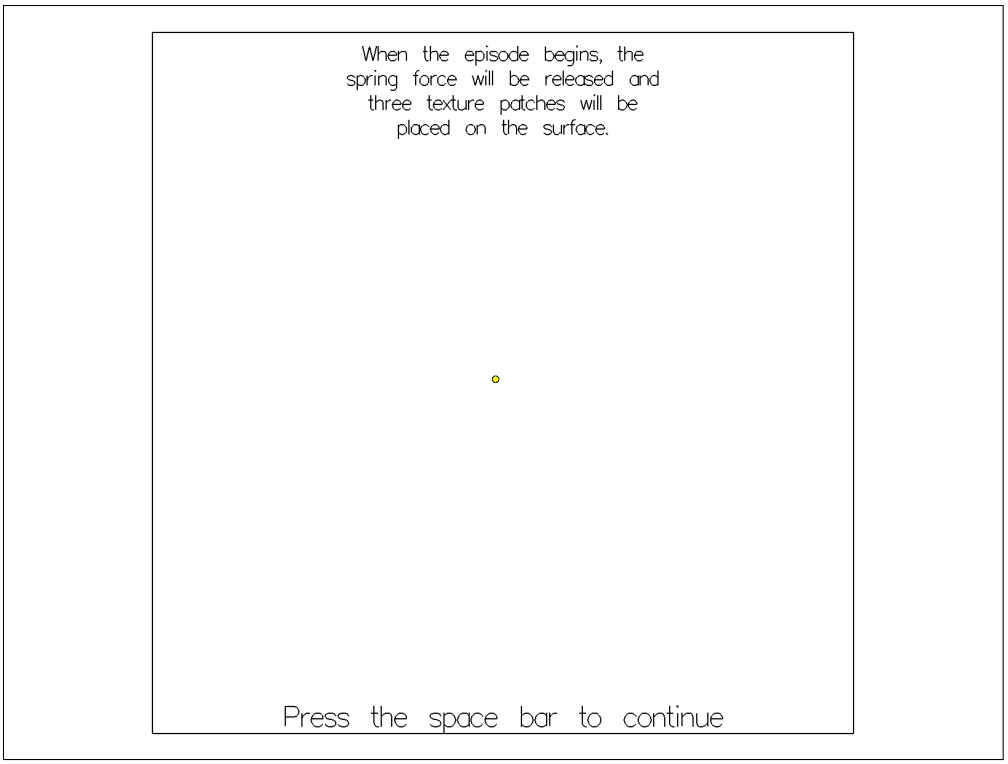

In the search task you'll be doing, you'll be trying to find patches that feel like the blue patch. Scrub the patches again to make sure you can recognize the blue patch's texture
Press the space bar to continue

(o)

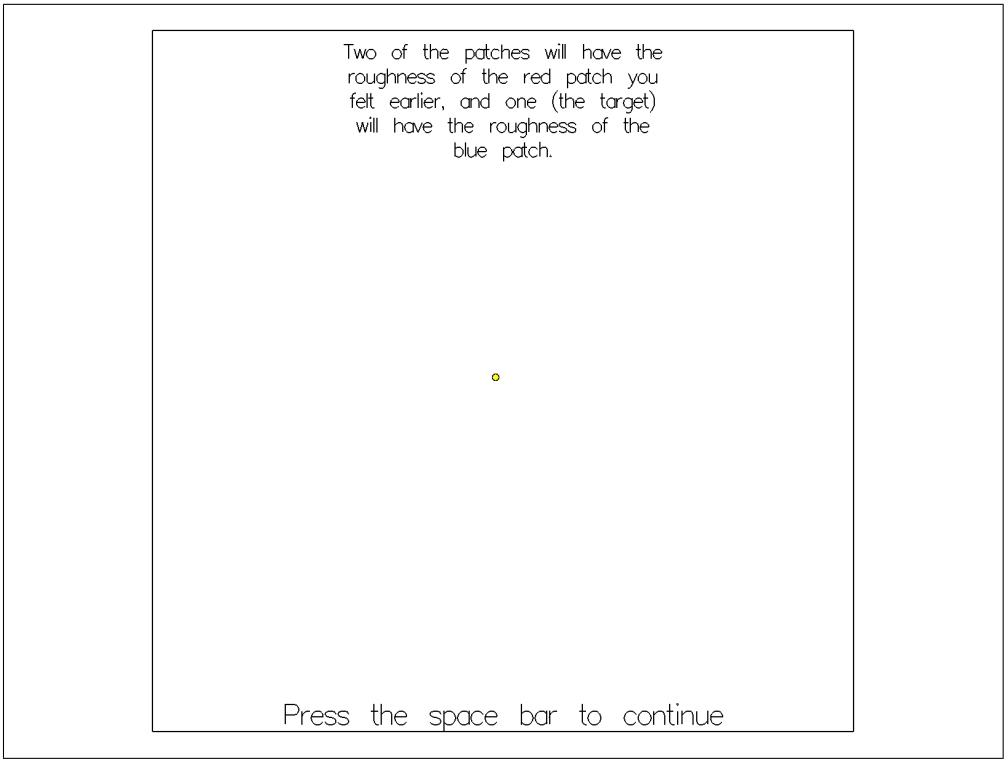

Now I'll explain the search task. Each episode of the task will start with the stylus held (by the spring force) in the centre of the workspace, just above the surface.
Press the space bar to continue

(p)

Figure A.1

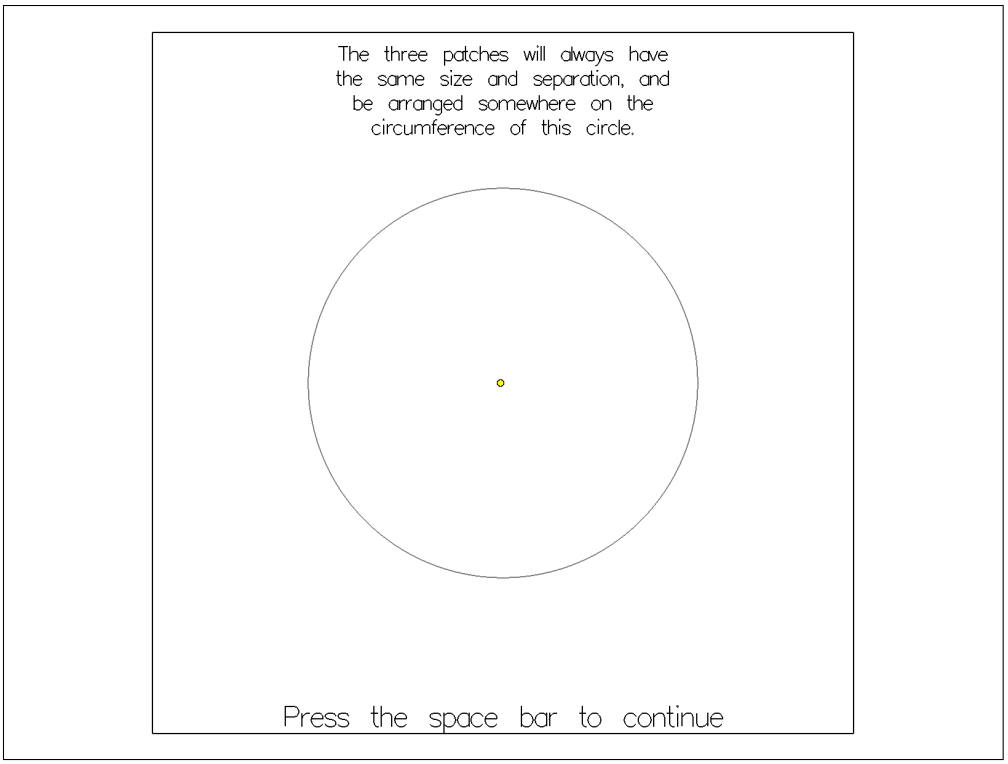

When the episode begins, the spring force will be released and three texture patches will be placed on the surface.
Press the space bar to continue

(q)

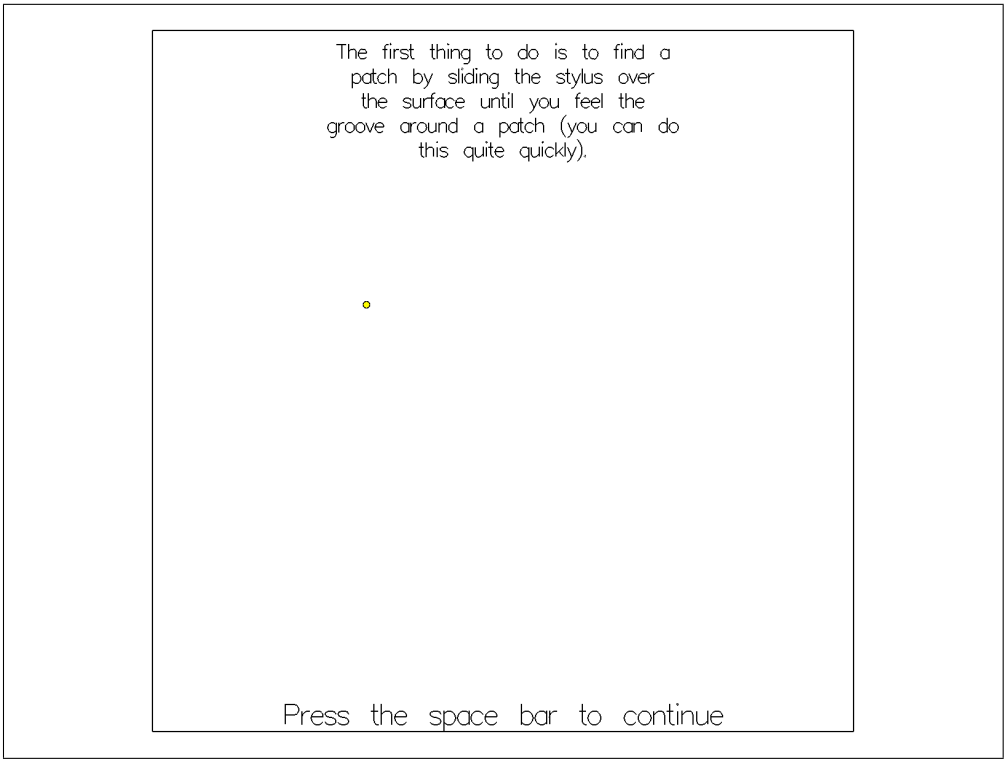

Two of the patches will have the roughness of the red patch you felt earlier, and one (the target) will have the roughness of the blue patch.
Press the space bar to continue

(r)

Figure A.1

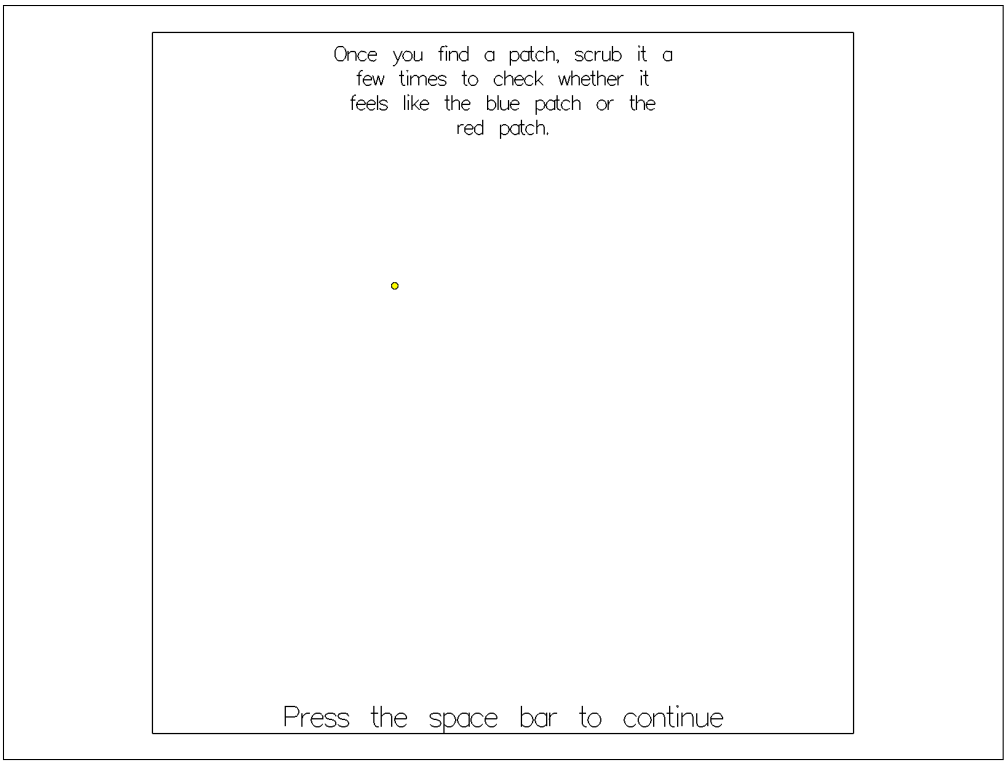

The three patches will always have the same size and separation, and be arranged somewhere on the circumference of this circle.
Press the space bar to continue

(s)

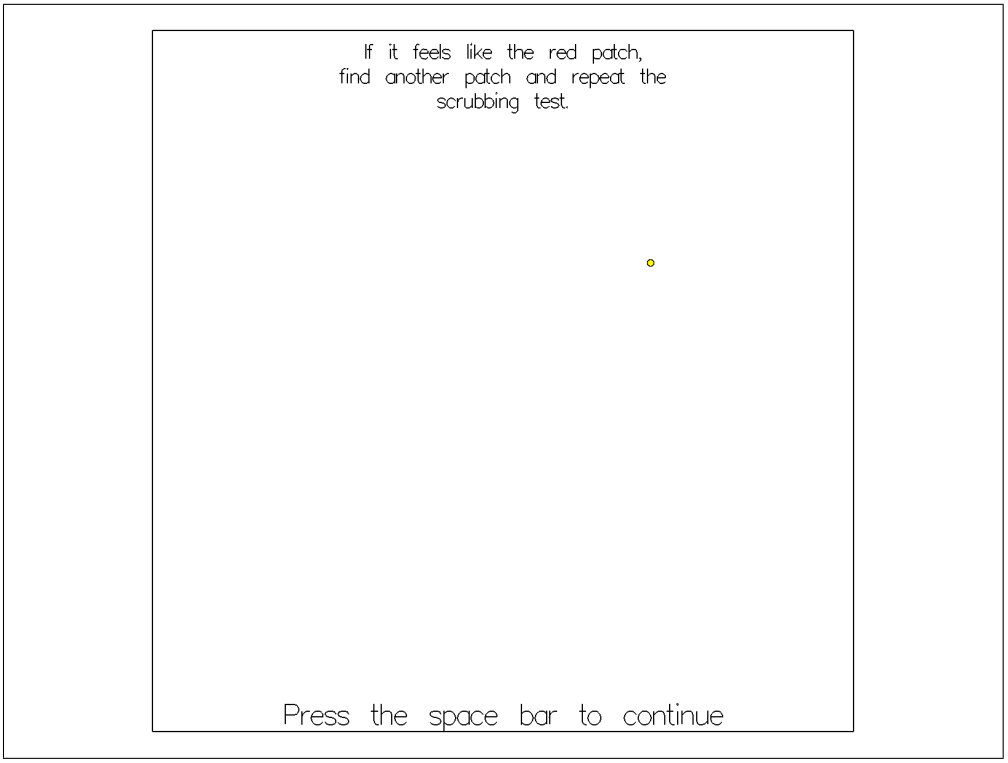

The first thing to do is to find a patch by sliding the stylus over the surface until you feel the groove around a patch (you can do this quite quickly).
Press the space bar to continue

(t)

Figure A.1

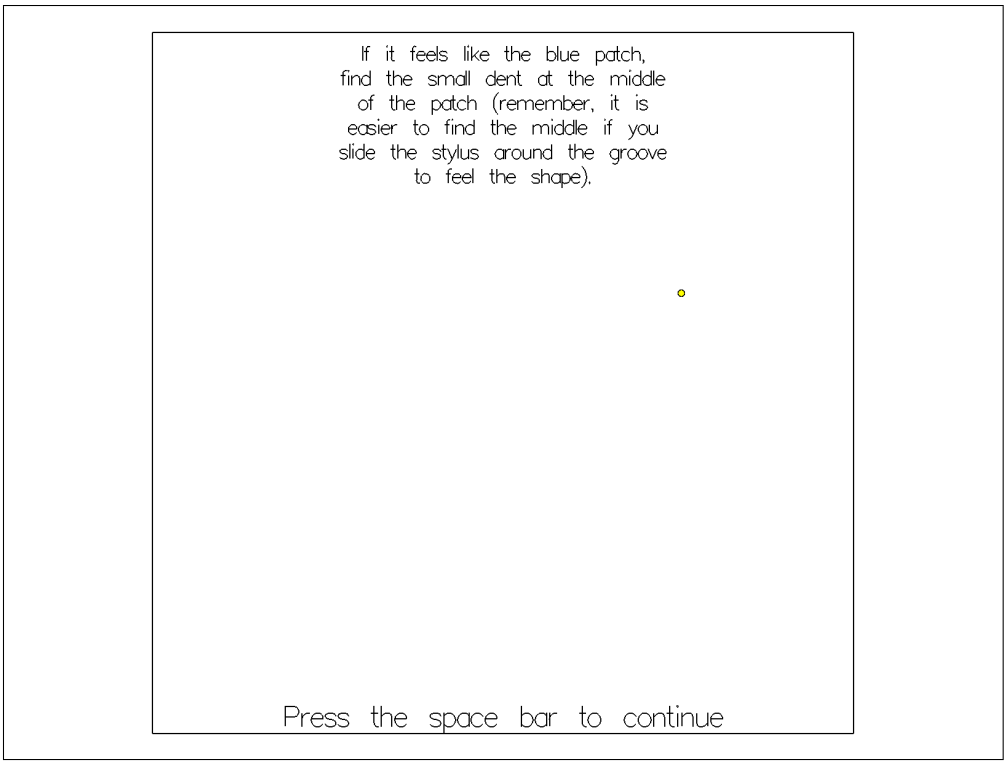

Once you find a patch, scrub it a few times to check whether it feels like the blue patch or the red patch.
Press the space bar to continue

(u)

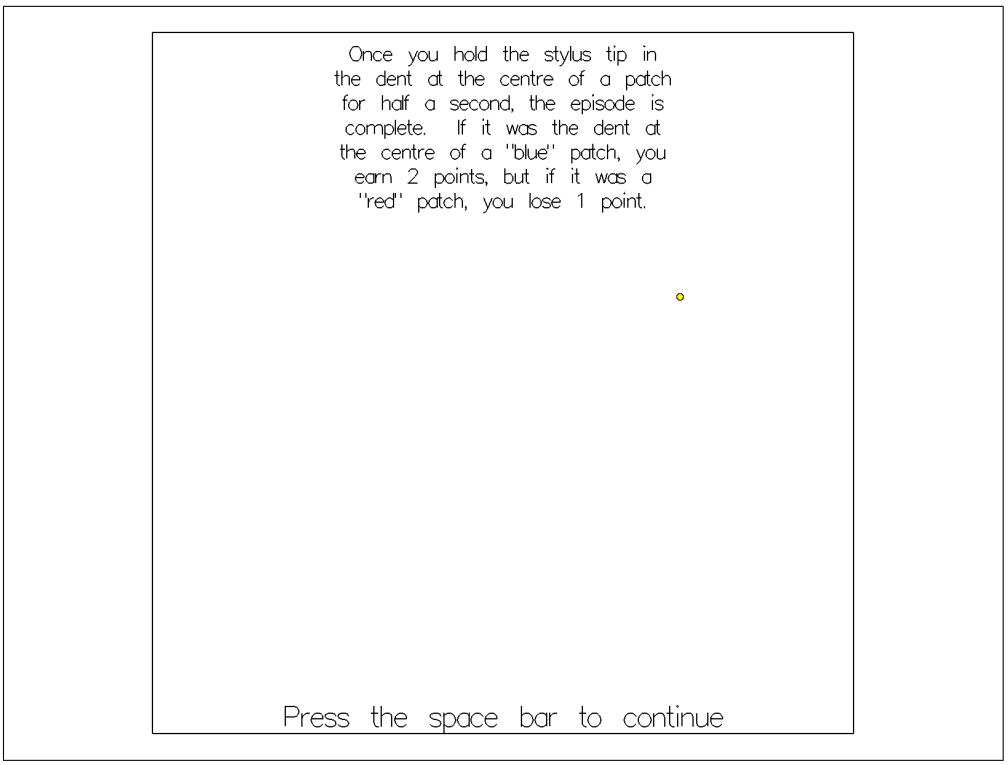

If it feels like the red patch, find another patch and repeat the scrubbing test.
Press the space bar to continue

(v)

Figure A.1

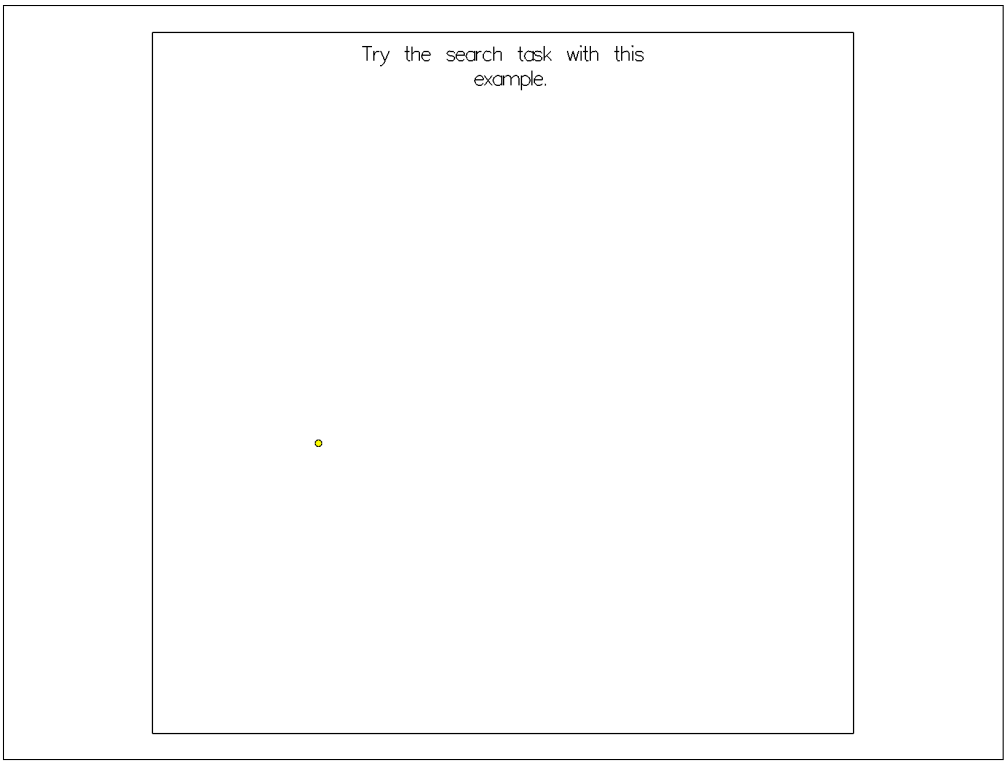

If it feels like the blue patch,
find the small dent at the middle
of the patch (remember, it is
easier to find the middle if you
slide the stylus around the groove
to feel the shape).
Press the space bar to continue

(w)

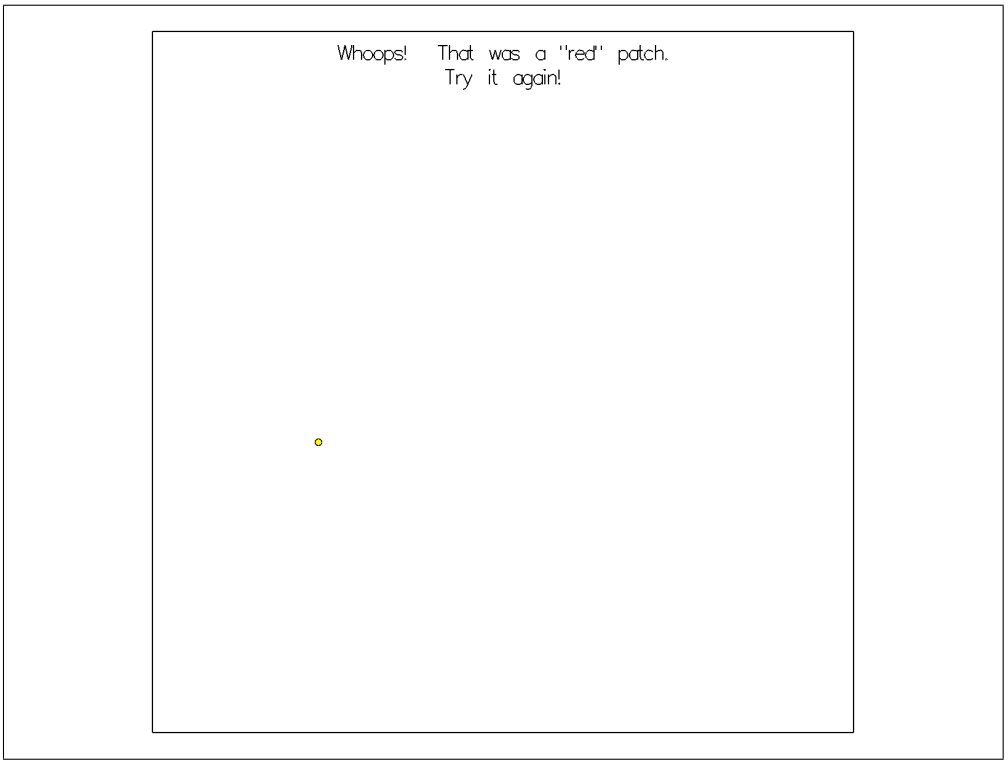

Once you hold the stylus tip in
the dent at the centre of a patch
for half a second, the episode is
complete. If it was the dent at
the centre of a "blue" patch, you
earn 2 points, but if it was a
"red" patch, you lose 1 point.
Press the space bar to continue

(x)

Figure A.1

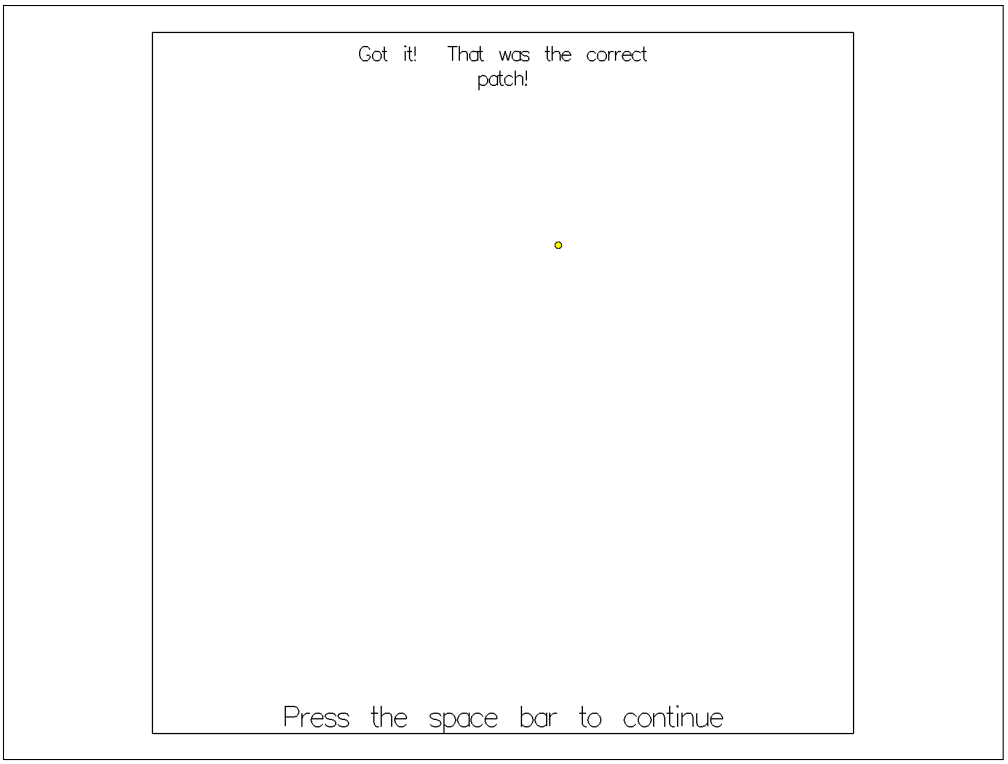


(y) If the subject selects the wrong (red) patch, the next screen is (z); otherwise the next screen is (aa).

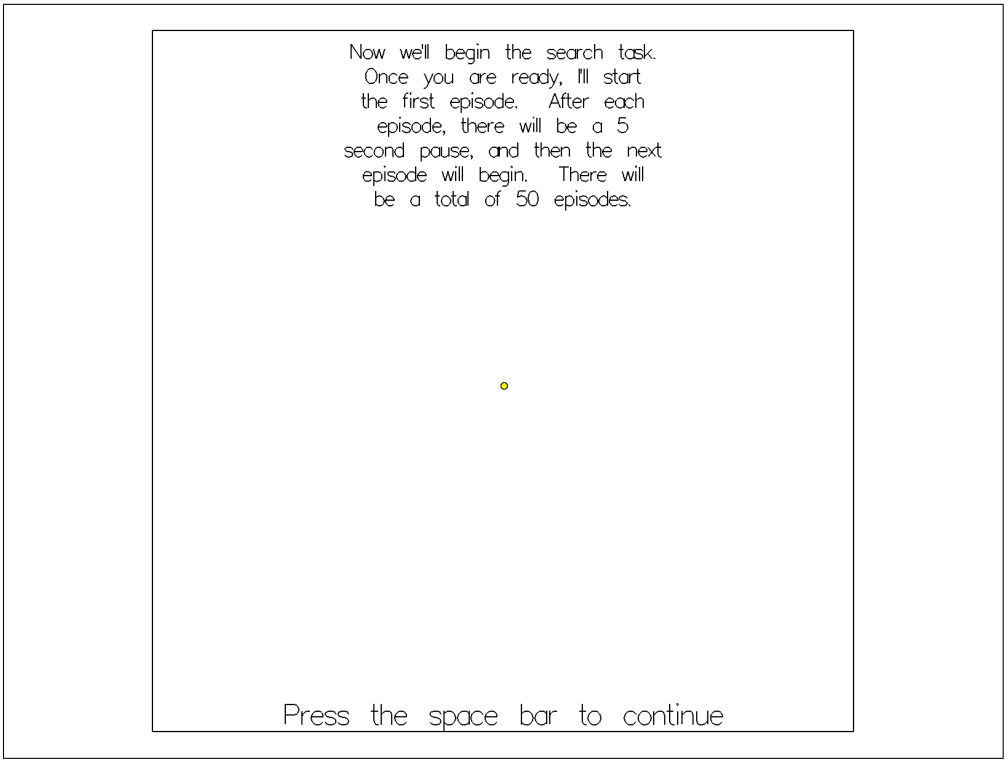


(z) After this screen, the subject is taken back to (y).

Figure A.1



(aa)



(ab)

Figure A.1

Bibliography

Gemma A. Calvert, Charles Spence, and Barry E. Stein, editors. *The Handbook of Multisensory Processes*. The MIT Press, 2004. 25, 34, 53

Timothy Edmunds, S. Muthukrishnan, Subarna Sadhukhan, and Shinjiro Sueda. MoDB: Database system for synthesizing human motion. In *ICDE '05: Proceedings of the 21st International Conference on Data Engineering*, pages 1131–1132. IEEE Computer Society, Washington, DC, USA, 2005. ISBN 0-7695-2285-8. doi: 10.1109/ICDE.2005.89. 18

Timothy Edmunds and Dinesh K. Pai. An event architecture for distributed interactive multisensory rendering. In *ISMAR 2006: IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 197–202. IEEE Computer Society, Los Alamitos, CA, USA, 2006. doi: 10.1109/ISMAR.2006.297814. iii

Timothy Edmunds and Dinesh K. Pai. Perceptual rendering for learning haptic skills. In *HAPTICS 2008: IEEE International symposium on Haptic Interfaces for Virtual Environments and Teleoperator Systems*, pages 225–230. IEEE Computer Society, 2008. ISBN 978-1-4244-2005-6. doi: 10.1109/HAPTICS.2008.4479948. iii

Timothy Edmunds and Dinesh K. Pai. Perceptually augmented simulator design through decomposition, 2009. To appear in *WHC '09: Third Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*. iii

Jesse D. Hwang, Michael D. Williams, and Günter Niemeyer. Toward event-based haptics: Rendering contact using open-loop force pulses. In *HAPTICS '04: Proceedings of the 12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 24–31. 2004. ISBN 0-7695-2112-6. doi: 10.1109/HAPTIC.2004.1287174. 26

Roland. S. Johansson. Sensory control of dexterous manipulation in humans. In Alan M. Wing, Patrick Haggard, and J. Randall Flanagan, editors, *Hand and Brain: The Neurophysiology and Psychology of Hand Movements*, pages 381–414. Academic Press, 1996. 25, 26

Danny M. Kaufman, Timothy Edmunds, and Dinesh K. Pai. Fast frictional dynamics for rigid bodies. In *SIGGRAPH '05: ACM SIGGRAPH 2005*

Papers, pages 946–956. ACM Press, New York, NY, USA, 2005. doi: 10.1145/1186822.1073295. 18

Katherine J. Kuchenbecker, Jonathan Fiene, and Günter Niemeyer. Improving contact realism through event-based haptic feedback. *IEEE Transactions on Visualization and Computer Graphics*, 12(2):219–230, March–April 2006. ISSN 1077-2626. doi: 10.1109/TVCG.2006.32. 26

Susan J. Lederman and Roberta L. Klatzky. Multisensory texture perception. In Calvert et al. [2004], pages 107–122. 25

Gavan Lintern. An informational perspective on skill transfer in human-machine systems. *Human Factors*, 33(3):251–266, June 1991. ISSN 0018-7208. 26

John E. Lloyd and Dinesh K. Pai. Robotic mapping of friction and roughness for reality-based modeling. In *IEEE International Conference on Robotics and Automation*, pages 1884–1890. 2001. 25

Henry W. Mertens. Perception of runway image shape and approach angle magnitude by pilots in simulated night landing approaches. *Aviation, Space, and Environmental Medicine*, 52(7):373–386, July 1981. 26

Allison M. Okamura, Mark R. Cutkosky, and Jack Tigh Dennerlein. Reality-based models for vibration feedback in virtual environments. *IEEE/ASME Transactions on Mechatronics*, 6(3):245–252, September 2001. ISSN 1083-4435. doi: 10.1109/3516.951362. 26

Gabriel Robles-De-La-Torre and Vincent Hayward. Force can overcome object geometry in the perception of shape through active touch. *Nature*, 412:445–448, 2001. doi: 10.1038/35086588. 25

S. E. Salcudean and T. D. Vlaar. On the emulation of stiff walls and static friction with a magnetically levitated input/output device. *ASME Journal of Dynamic Systems, Measurement and Control*, 119(1):127–132, March 1997. ISSN 0022-0434. 26

M. A. Srinivasan, G. L. Beauregard, and D. L. Brock. The impact of visual information on the haptic perception of stiffness in virtual environments. In *ASME 1996: Proceedings of the ASME Dynamics Systems and Control Division*, volume 58, pages 555–559. ASME, 1996. 25

Barry E. Stein and M. Alex Meredith. *The Merging of the Senses*. Cognitive Neuroscience. The MIT Press, 1993. 25

Dennis C. Wightman and Gavan Lintern. Part-task training for tracking and manual control. *Human Factors*, 27(3):267–283, June 1985. 26

Vita

Timothy Edmunds

1996–2001 B. Sc. from an undergraduate university

2007–08 Bevier Fellow, Department of Computer Science, Rutgers University

2002–07 Graduate/Teaching assistant, Department of Computer Science, Rutgers University

2001–02 Project Manager, Real World Company Ltd.

Publications

Timothy Edmunds and Dinesh K. Pai. Perceptually Augmented Simulator Design Through Decomposition, 2009. To appear in *WHC '09: Third Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*.

Timothy Edmunds and Dinesh K. Pai. Perceptual Rendering for Learning Haptic Skills. In *HAPTICS 2008: IEEE International symposium on Haptic Interfaces for Virtual Environments and Teleoperator Systems*, pages 225–230. IEEE Computer Society, 2008. ISBN 978-1-4244-2005-6. doi: 10.1109/HAPTICS.2008.4479948.

Bethany R. Leffler, Michael L. Littman, and Timothy Edmunds. Efficient Reinforcement Learning with Relocatable Action Models. In *AAAI-07: Proceedings of the Twenty-Second Conference on Artificial Intelligence*, pages 572–577. The AAAI Press, Menlo Park, CA, USA, 2007.

Timothy Edmunds and Dinesh K. Pai. An Event Architecture for Distributed Interactive Multisensory Rendering. In *ISMAR 2006: IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 197–202. IEEE Computer Society, Los Alamitos, CA, USA, 2006. doi: 10.1109/ISMAR.2006.297814.

Emanuele Ruffaldi, Dan Morris, Timothy Edmunds, Federico Barbagli, and Dinesh K.Pai. Standardized Evaluation of Haptic Rendering Systems. In *HAPTICS '06: Proceedings of the Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pages 255–232. IEEE Computer Society, Washington, DC, USA, 2006. ISBN 1-4244-0226-3. doi: 10.1109/HAPTICS.2006.175.

Timothy Edmunds, S. Muthukrishnan, Subarna Sadhukhan, and Shinjiro Sueda. MoDB: Database System for Synthesizing Human Motion. In *ICDE '05: Proceedings of the 21st International Conference on Data Engineering*, pages 1131–1132. IEEE Computer Society, Washington, DC, USA, 2005. ISBN 0-7695-2285-8. doi: 10.1109/ICDE.2005.89.

Danny M. Kaufman, Timothy Edmunds, and Dinesh K. Pai. Fast Frictional Dynamics for Rigid Bodies. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*, pages 946–956. ACM Press, New York, NY, USA, 2005. doi: 10.1145/1186822.1073295.

Danny M. Kaufman, Timothy Edmunds, and Dinesh K. Pai. Implementation of Fast Frictional Dynamics for Rigid Bodies. In *SIGGRAPH '05: ACM SIGGRAPH 2005 Sketches*, page 78. ACM Press, New York, NY, USA, 2005. doi: 10.1145/1187112.1187205.

T. Edmunds, D. Kaufman, P.G. Kry, D.K. Pai, S. Sadhukhan, S. Sueda, D. Wang, and Q. Wei. Interactive Character Animation with Vision. In *SCA '04: Poster Proceedings of the 2004 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pages 42–43. 2004.