Abstract

We have been exploiting the capabilities of multimedia tools to combine visual and audio images to create a multilingual reference tool for architecture students. Our students work in both Cantonese and English, learning incomplete architectural vocabularies in both languages. We set out to create a tool which can be used by students as a reference tool as well as a teaching tool. In response to this need, we have created a multimedia lexicon using three-dimensional models of buildings with which the user can interact, identify elements and hear and see the correct term for the elements identified in the language of their choice. The system can handle any number of language combinations or building models. This paper describes a system currently under development which has gone through two iterations to explore the issues involved.

Introduction

Our goal is simple - to address the need to bring our students’ technical vocabulary up to a reasonably consistent level in two languages. This is a niggling problem which is inevitable in a multilingual society - the students come to us from backgrounds in which they use at least two languages, one at home (Cantonese) and another in which they are taught (English). Add to this a third complexity – Cantonese is a minority language (only 200 million people speak it or a language closely related) so this minority must learn the official national tongue (Putunghua) of our future landlords (China).

Under these conditions, we try to teach architecture and its vocabulary. We have the potential that a student does not know a word in three languages (at least) and probably needs to know it in two. Obviously, the tutors can be ignorant of up to two of the terms while they try to teach the third. Add to this that not one of us seems to have a full set of terms in two languages - we all have a good grasp of one language and a knowledge of another (to varying degrees). Thus, not even the tutors can consistently help the students become familiar with the diversity of terms we need to know.

This situation demands the development of a tool to collect, co-ordinate and retain terms for easy visual access by the user. A visual / audio lexicon seems to be the answer. The lexicon will need to be easily used, readily accessible and easy to expand (by adding words or languages).

There are a number of multilingual visual dictionaries in existence (Corbell 1988, Oxford-Duden 1981) which include some entries related to buildings and architecture, and one monolingual one that deals exclusively with architectural elements (Calloway 1991).
Printed media, though, does not address an important aspect of dealing with a non-phonetic script such as Chinese - how do you pronounce the word? A multimedia environment is therefore the answer, and a multilingual visual lexicon the application to build using such tools.

System design

The goal is to produce a system which is flexible enough to handle any number of languages, any kind of building model and run on a variety of hardware platforms.
The user should be able to specify the first (or base) language for their interaction – this is the language to which all other terms will be compared. They can then specify a second language they wish to learn. Thus, each query is answered by a bilingual response - the building component is identified as text in the base language as well as the second language and the user hears the spoken term in the second language.

The two significant tasks are (1) building the digital model and (2) gathering the terms.

Building the model

The system requires a full 3D model which can be disassembled into component pieces. We built the model in Autocad and translated the DXF file into OpenInventor format which is used by Showcase and Inventor. The model is nested in hierarchical form, using a syntactic model of the building as our guide. The hierarchy chosen is starts from the major structural and cladding elements of the building and progressively refines these to their component elements.

One significant draw back of this nesting is that it dictates the ways in which the user can explore the building and learn the names. We are, in effect, enforcing one view of cataloguing the components of the building which we recognise may not always be the most relevant presentation. At this point in our explorations, we accepted the compromise for expediency. This nesting structure fits the grouping concepts used in OpenInventor and Autocad (blocks). Obviously a relational object database would provide greater flexibility in organising the data.
Each graphic element is labelled with a mnemonic name to ease the task of cataloguing the building components later. The convention for naming audio and text files adopts a scheme which divides a filename in three parts.

"[mnemonic name]-[language label].[media type]"

The root of a filename contains the mnemonic name of the sub-component it represent as well as its language, which are separated by a hyphen. (e.g. douKung-e is dou kung in English, column-c stands for column in Chinese. We are labelling the languages with "c" for Chinese, "d" for German (Deutsch), "e" for English, "j" for Japanese and "k" for Korean. Obviously this is limiting (we can only have 26 languages, a - z, but this is a temporary work around).

The extension of a filename tells what the media type of the file is. (e.g. douKung-e.au is the audio file for dou kung in English, column-c.txt is the Chinese text for column in Big-5 encoded form.) These extensions are the labelling conventions.
for media type in the computer science paradigm.
Adding a language is straightforward. The information of which components link to which audio and text files are stored in a file. The file adopted the SGML (Standard General Markup Language) format, with its own rule for DTD (Document Type Definition). An Entry of a sub-component in the file will look like this:

```xml
<INSTANCE>
  <NAME>douKung</NAME>
  <LABEL>
    <LANG1>douKung-e.txt</LANG1>
    <LANG2>douKung-c.txt</LANG2>
  </LABEL>
  <AUDIO>
    <LANG1>douKung-e.au</LANG1>
    <LANG2>douKung-c.au</LANG2>
  </AUDIO>
</INSTANCE>
```

In the DTD, every sub-component is marked by an INSTANCE tag. The NAME tag marks the mnemonic name of the sub-component being represented, this field is the primary key for searching in the database. The LABEL tag encloses the names of text files in different languages that represent the sub-component. The AUDIO tag encloses filenames for audio, similar to the LABEL tag. To add a new language, you just insert filenames between those media tags and enclose them by <LANG?> and </LANG?>, where ? is the number of languages the system supported after the addition of the new language. The files referenced by the newly added filenames should contain media that are said in the new language. At the moment, the maximum number of languages supported by the system simultaneously is 4, which can be increased later when necessary.

Gathering the technical terms

This is perhaps the most difficult aspect of compiling the system and one intentionally we have not fully addressed. We did not set out to meet lexical standards for our compilation – we are not creating a definitive dictionary of architectural terms. There are two problems here, those of translation and of oral traditions.

The translation of technical terms is always fraught with problems. We have tried as far as possible to rely upon standard dictionaries for our words (Oxford-Duden 1981) but these sources are necessarily inadequate since there are no exhaustive translations to English or other languages of the terms used in Chinese architecture. The second problem is just what do you call a component in a building in the Chinese tradition. This is a fundamental problem since there are no standard terms; one of our initial languages (Cantonese) is essentially a spoken language and building terms varying from region to region, culture to culture, within the language zone. In Hong Kong we see this on building sites where teams of immigrant workers from around this region work in closely related languages but use different terms to refer to the same components. Instead of attempting to be definitive in our lexicon, we have drawn upon standard texts (among which sometimes there are disagreements for terms in Chinese architecture) and sacrificed lexical accuracy for expediency of compilation; we are architects, after all, not linguists.

System configuration

The initial concept was to develop a system which is fully portable or readily usable on the Windows/PC platform since this is the most pervasive system. The tools available for development on the PC platform did not seem adequate for our initial work, so we elected instead to prototype the system on a Silicon Graphics machine using Inventor and SGML tools. Once a user interface model is established, we expect to port the system to the PC platform. Meanwhile, we can make the system available to students on the SGI network in use in our department.

The second version of the prototype was built on the SGI platform running IRIX 5.2. As the system was written using the OpenInventor Programming Toolkit and the 3D viewer’s user interface was based on Motif, it requires the OpenInventor executable environment as well as the Motif executable environment to run. The language support module was written in TCL/TK which communicates with the main system using the TCL-DP protocol, it also requires a TCL/TK interpreter with TCL-DP support. The system accepts standard X-Window fonts for displaying text in different languages. We use Big-5 encoding for Chinese and JIS-Kanji encoding to display Japanese. The system runs smoothly on an SGI Indigo R3000 as well as on an SGI Indigo² Extreme.

Reviewing the results

As mentioned above, we have developed two generations of the system. The first was developed by students as part of a course in our Masters of Architecture programme. This version of the system was developed using Showcase on an SGI system, using an existing CAD model of a temple. Showcase permitted the students to quickly package images and reference audio files within a consistent user interface. The drawback is that the user’s path is strictly controlled by the hierarchy – you are essentially looking at a multimedia slide show in which you can occasionally make choices in direction.
The first version worked well in exploring the questions of user interaction, but it was inflexible. While the first implementation looked polished, it suffered from an erratic user interface (reflecting the different people working on different stages). The Chinese text was scanned, rendering some of it difficult to read. It is impossible to add additional languages and difficult to add another building model.

A second implementation was developed to address some of these problems and to investigate the questions of file structure and multi-language control. This we achieved but with an unsatisfactory user interface arising from the different personalities of the various development tools employed. We wrestled with different ways of associating building components to permit different views of the building but failed to answer this need satisfactorily.

Conclusions and next steps

Cross-platform functionality is proving a problem since the development tools we need seem to be platform specific. We are still choosing to remain with the SGI platform as we have greater processing power for prototyping and multimedia manipulation.

Future exploration will be to develop a database approach to cataloguing and storing audio, character and graphic data. This should enhance the access control. We will also look into adding photographs as well as cross references to other buildings or historical information.

The greatest positive enhancement would be to add voice input. Existing input methods for 16-bit fonts (such as Chinese Big-5 and JIS-Kanji) are non-trivial, making it difficult to set up queries using these languages as base languages. To overcome this, the system should be able to accept verbal queries in Chinese—“Show me a dou-kung and tell me what it is called in French” or words to that effect.

Acknowledgements

This work was carried out with funding provided by the University of Hong Kong using resources of the Department of Architecture. The first version of the system was developed in 1994 by a student team: Chan Ming Yan, Cheung Kam Cheung, Choy Ngar Yee, Fung Siu Man, Ng Man Chuen. Research assistant Matchy Ma assisted in developing the second version of the system.

References


Corbell, J. C., 1988 Visual Dictionary, Hong Kong: Reader’s Digest Association Far East Limited