Design of a Tourist Driven Bandwidth Determined MultiModal Mobile Presentation System

Anthony Solon, Paul Mc Kevitt, and Kevin Curran

Intelligent Multimedia Research Group
School of Computing and Intelligent Systems, Faculty of Engineering
University of Ulster, Magee Campus, Northland Road, Northern Ireland, BT48 7JL, UK
{aj.solon,p.mckevitt,kj.curran}@ulster.ac.uk

Abstract. TeleMorph is a tourist information system which aims to dynamically generate multimedia presentations using output modalities that are determined by the bandwidth available on a mobile device's connection. This paper concentrates on the motivation for & issues surrounding such intelligent systems.

1 Introduction

Interfaces involving spoken or pen-based input, as well as the combination of both, are particularly effective for supporting mobile tasks, such as communications and personal navigation. Unlike the keyboard and mouse, both speech and pen are compact and portable. When combined, people can shift these input modes from moment to moment as environmental conditions change [1]. Implementing multimodal user interfaces on mobile devices is not as clear-cut as doing so on ordinary desktop devices. This is due to the fact that mobile devices are limited in many respects: memory, processing power, input modes, battery power, and an unreliable wireless connection with limited bandwidth. This project researches and implements a framework for Multimodal interaction in mobile environments taking into consideration fluctuating bandwidth. The system output is bandwidth dependent, with the result that output from semantic representations is dynamically morphed between modalities or combinations of modalities. With the advent of 3G wireless networks and the subsequent increased speed in data transfer available, the possibilities for applications and services that will link people throughout the world who are connected to the network will be unprecedented. One may even anticipate a time when the applications and services available on wireless devices will replace the original versions implemented on ordinary desktop computers. Some projects have already investigated mobile intelligent multimedia systems, using tourism in particular as an application domain. [2] is one such project which analysed and designed a position-aware speech-enabled handheld tourist information system for Aalborg in Denmark. This system is position and direction aware and uses these abilities to guide a tourist on a sight seeing tour. In TeleMorph bandwidth will primarily determine the modality/modalities utilised in the output presentation, but also factors such as device constraints, user goal and user situationalisation will be taken into consideration. A provision will also be integrated which will allow users to choose their preferred modalities. The main point to note about these systems is that current mobile intelligent multimedia systems fail to take into consideration network constraints and especially the bandwidth available when

A. Karmouch, L. Korba, and E. Madeira (Eds.): MATA 2004, LNCS 3284, pp. 331–338, 2004.

transforming semantic representations into the multimodal output presentation. If the bandwidth available to a device is low then it's obviously inefficient to attempt to use video or animations as the output on the mobile device. This would result in an interface with depreciated quality, effectiveness and user acceptance. This is an important issue as regards the usability of the interface. Learnability, throughput, flexibility and user-attitude are the four main concerns affecting the usability of any interface. In the case of the previously mentioned scenario (reduced bandwidth => slower/inefficient output) the throughput of the interface is affected and as a result the user's attitude also. This is only a problem when the required bandwidth for the output modalities exceeds that which is available; hence, the importance of choosing the correct output modality/modalities in relation to available resources. The next section deals with related multi-modal systems. The following section presents TeleMorph while section 3 presents an overview of other Multi-modal systems. Section 4 concludes.

2 TeleMorph

The focus of the TeleMorph project is to create a system that dynamically morphs between output modalities depending on available network bandwidth. The aims entail the following objectives which include receiving and interpreting questions from the user; Mapping questions to multimodal semantic representation; matching multimodal representation to database to retrieve answer; mapping answers to multimodal semantic representation; querying bandwidth status and generating multimodal presentation based on bandwidth data. The domain chosen as a test bed for TeleMorph is *e*Tourism. The system to be developed called TeleTuras is an interactive tourist information aid. It will incorporate route planning, maps, points of interest, spoken presentations, graphics of important objects in the area and animations. The main focus will be on the output modalities used to communicate this information and also the effectiveness of this communication. The tools that will be used to implement this system are detailed in the next section. TeleTuras will be capable of taking input queries in a variety of modalities whether they are combined or used individually. Queries can also be directly related to the user's position and movement direction enabling questions/commands such as "Where is the Leisure Center?", "Take me to the Council Offices" and "What buildings are of interest in this area?".

J2ME (Java 2 Micro Edition) is an ideal programming language for developing TeleMorph, as it is the target platform for the Java Speech API (JSAPI) [3]. The JSAPI enables the inclusion of speech technology in user interfaces for Java applets and applications. The Java Speech API Markup Language [4] and the Java Speech API Grammar Format [4] are companion specifications to the JSAPI. JSML (currently in beta) defines a standard text format for marking up text for input to a speech synthesiser. JSGF version 1.0 defines a standard text format for providing a grammar to a speech recogniser. JSAPI does not provide any speech functionality itself, but through a set of APIs and event interfaces, access to speech functionality provided by supporting speech vendors is accessible to the application. As it is inevitable that a majority of tourists will be foreigners it is necessary that TeleTuras can process multilingual speech recognition and synthesis. To support this an IBM implementation of JSAPI "speech for Java" will be utilised. It supports US&UK English, French, German, Italian, Spanish, and Japanese. To incorporate the navigation aspect of the pro-

posed system a positioning system is required. The GPS (Global Positioning System) [2] will be employed to provide the accurate location information necessary for a LBS (Location Based Service). The User Interface (UI) defined in J2ME is logically composed of two sets of APIs, High-level UI API which emphasises portability across different devices and the Low-level UI API which emphasises flexibility and control. TeleMorph will use a dynamic combination of these in order to provide the best solution possible. Media Design takes the output information and morphs it into relevant modality/modalities depending on the information it receives from the Server Intelligent Agent regarding available bandwidth, whilst also taking into consideration the Cognitive Load Theory as described earlier. Media Analysis receives input from the Client device and analyses it to distinguish the modality types that the user utilised in their input. The Domain Model, Discourse Model, User Model, GPS and WWW are additional sources of information for the Multimodal Interaction Manager that assist it in producing an appropriate and correct output presentation. The Server Intelligent Agent is responsible for monitoring bandwidth, sending streaming media which is morphed to the appropriate modalities and receiving input from client device & mapping to multimodal interaction manager. The Client Intelligent Agent is in charge of monitoring device constraints e.g. memory available, sending multimodal information on input to the server and receiving streamed multimedia.

2.1 Data Flow of TeleMorph

The *Networking API* sends all input from the client device to the TeleMorph server. Each time this occurs, the *Device Monitoring* module will retrieve information on the client device's status and this information is also sent to the server. On input the user can make a multimodal query to the system to stream a new presentation which will consist of media pertaining to their specific query. TeleMorph will receive requests in the *Interaction Manager* and will process requests via the *Media Analysis* module which will pass semantically useful data to the *Constraint Processor* where modalities suited to the current network bandwidth (and other constraints) will be chosen to represent the information. The presentation is then designed using these modalities by the *Presentation Design* module. The media are processed by the *Media Allocation* module and following this the complete multimodal Synchronised Multimedia Integration Language (SMIL) [5] presentation is passed to the *Streaming Server* to be streamed to the client device. A user can also input particular modality/cost choices on the TeleMorph client. In this way the user can morph the current presentation they are receiving to a presentation consisting of specific modalities which may be better suited their current situation (driving/walking) or environment (work/class/pub). The Mobile Client's Output Processing module will process media being streamed to it across the wireless network and present the received modalities to the user in a synchronised fashion. The Input Processing module on the client will process input from the user in a variety of modes. This module will also be concerned with timing thresholds between different modality inputs. In order to implement this architecture for initial testing, a scenario will be set up where switches in the project code will simulate changing between a variety of bandwidths. To implement this, TeleMorph will draw on a database which will consist of a table of bandwidths ranging from those available in 1G, 2G, 2.5G (GPRS) and 3G networks. Each bandwidth value will have access to related information on the modality/combinations of modalities that can be streamed efficiently at that transmission rate.

2.2 Client Output

Output on thin client devices connected to TeleMorph will primarily utilise a SMIL media player which will present video, graphics, text and speech to the end user of the system. The J2ME Text-To-Speech (TTS) engine processes speech output to the user. An autonomous agent will be integrated into the TeleMorph client for output as they serve as an invaluable interface agent to the user as they incorporate modalities that are the natural modalities of face-to-face communication among humans. A SMIL media player will output audio on the client device. This audio will consist of audio files that are streamed to the client when the necessary bandwidth is available.

2.3 Client Input

The TeleMorph client will allow for speech recognition, text and haptic deixis (touch screen) input. A speech recognition engine will be reused to process speech input from the user. Text and haptic input will be processed by the J2ME graphics API. Speech recognition in TeleMorph resides in *Capture Input* as illustrated in Figure 1.

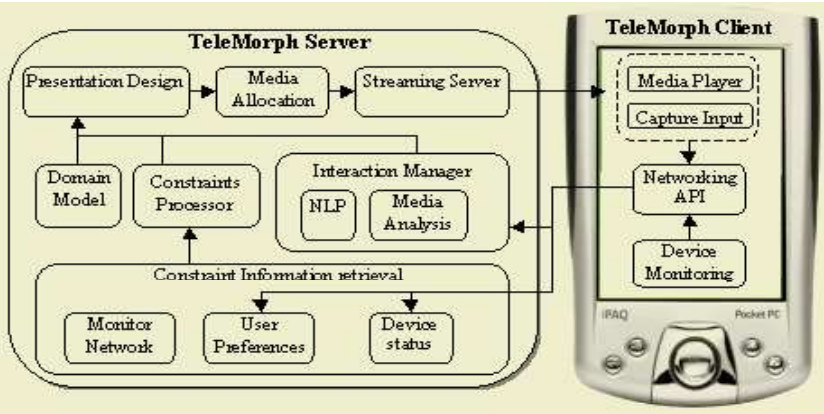

Fig. 1. Modules within TeleMorph

The Java Speech API Mark-up Language[1] defines a standard text format for marking up text for input to a speech synthesiser. As mentioned before JSAPI does not provide any speech functionality itself, but through a set of APIs and event interfaces, access to speech functionality (provided by supporting speech vendors) is accessible to the application. For this purpose IBM's implementation of JSAPI "speech for Java" is adopted for providing multilingual speech recognition functionality. This implementation of the JSAPI is based on ViaVoice, which will be positioned remotely in the *Interaction Manager* module on the server. The relationship between the JSAPI speech recogniser (in the *Capture Input* module in Figure 1) on the client and ViaVoice (in the *Interaction Manager* module in Figure 1) on the server is necessary as speech recognition is computationally too heavy to be processed on a thin client. After the ViaVoice speech recogniser has processed speech which is input to the client device, it will also need to be analysed by an *NLP* module to assess its semantic content. A reusable tool to do this is yet to be decided upon to complete this task.

[1] http://java.sun.com/products/java-media/speech/

Possible solutions for this include adding an additional NLP component to ViaVoice; or perhaps reusing other natural understanding tools such as PC-PATR [6] which is a natural language parser based on context-free phrase structure grammar and unifications on the feature structures associated with the constituents of the phrase structure rules.

2.4 Graphics

The User Interface (UI) defined in J2ME is logically composed of two sets of APIs, High-level UI API which emphasises portability across different devices and the Low-level UI API which emphasises flexibility and control. The portability in the high-level API is achieved by employing a high level of abstraction. The actual drawing and processing user interactions are performed by implementations. Applications that use the high-level API have little control over the visual appearance of components, and can only access high-level UI events. On the other hand, using the low-level API, an application has full control of appearance, and can directly access input devices and handle primitive events generated by user interaction. However the low-level API may be device-dependent, so applications developed using it will not be portable to other devices with a varying screen size. TeleMorph uses a combination of these to provide the best solution possible. Using these graphics APIs, TeleMorph implements a *Capture Input* module which accepts text from the user. Also using these APIs, haptic input is processed by the *Capture Input* module to keep track of the user's input via a touch screen, if one is present on the device. User preferences in relation to modalities and cost incurred are managed by the *Capture Input* module in the form of standard check boxes and text boxes available in the J2ME high level graphics API.

2.5 Networking

Networking takes place using sockets in the *J2ME Networking API* module as shown in Figure 1 to communicate data from the *Capture Input* module to the *Media Analysis* and *Constraint Information Retrieval* modules on the server. Information on client device constraints will also be received from the *Device Monitoring* module to the *Networking API* and sent to the relevant modules within the *Constraint Information Retrieval* module on the server. Networking in J2ME has to be very flexible to support a variety of wireless devices and has to be device specific at the same time. To meet this challenge, the Generic Connection Framework (GCF) is incorporated into J2ME. The idea of the GCF is to define the abstractions of the networking and file input/output as generally as possible to support a broad range of devices, and leave the actual implementations of these abstractions to the individual device manufacturers. These abstractions are defined as Java interfaces. The device manufacturers choose which one to implement based on the actual device capabilities.

2.6 TeleMorph Server-Side

SMIL is utilised to form the semantic representation language in TeleMorph and will be processed by the *Presentation Design* module in Figure 1. The HUGIN[2] develop-

[2] HUGIN (2003) http://www.hugin.com/

ment environment allows TeleMorph to develop its decision making process using Causal Probabilistic Networks which will form the *Constraint Processor* module as portrayed in Figure 1. The ViaVoice speech recognition software resides within the *Interaction Manager* module. On the server end of the system Darwin streaming server[3] is responsible for transmitting the output presentation from the TeleMorph server application to the client device's *Media Player*.

2.6.1 SMIL Semantic Representation

The XML based Synchronised Multimedia Integration Language (SMIL) language [5] forms the semantic representation language of TeleMorph used in the *Presentation Design* module as shown in Figure 1. TeleMorph designs SMIL content that comprises multiple modalities that exploit currently available resources fully, whilst considering various constraints that affect the presentation, but in particular, bandwidth. This output presentation is then streamed to the *Media Player* module on the mobile client for displaying to the end user. TeleMorph will constantly recycle the presentation SMIL code to adapt to continuous and unpredictable variations of physical system constraints (e.g. fluctuating bandwidth, device memory), user constraints (e.g. environment) and user choices (e.g. streaming text instead of synthesised speech). In order to present the content to the end user, a SMIL media player needs to be available on the client device. A possible contender to implement this is MPEG-7, as it describes multimedia content using XML.

2.6.2 TeleMorph Reasoning – CPNs/BBNs

Causal Probabilistic Networks aid in conducting reasoning and decision making within the *Constraints Processor* module (see Figure 1). In order to implement Bayesian Networks in TeleMorph, the HUGIN [7] development environment is used. HUGIN provides the necessary tools to construct Bayesian Networks. When a network has been constructed, one can use it for entering evidence in some of the nodes where the state is known and then retrieve the new probabilities calculated in other nodes corresponding to this evidence. A Causal Probabilistic Network (CPN)/Bayesian Belief network (BBN) is used to model a domain containing uncertainty in some manner. It consists of a set of nodes and a set of directed edges between these nodes. A Belief Network is a Directed Acyclic Graph (DAG) where each node represents a random variable. Each node contains the states of the random variable it represents and a conditional probability table (CPT) or, in more general terms, a conditional probability function (CPF). The CPT of a node contains probabilities of the node being in a specific state given the states of its parents. Edges reflect cause-effect relations within the domain. These effects are normally not completely deterministic (e.g. disease -> symptom). The strength of an effect is modelled as a probability.

2.6.3 JATLite Middleware

As TeleMorph is composed of several modules with different tasks to accomplish, the integration of the selected tools to complete each task is important. To allow for this a middleware is required within the *TeleMorph Server* as portrayed in figure 1. One such middleware is JATLite [8] which was developed by the Stanford University. JATLite provides a set of Java packages which makes it easy to build multi-agent systems using Java. As an alternative to the JATLite middleware The Open Agent

[3] http://developer.apple.com/darwin/projects/darwin/

Architecture (OAA) [9] could be used. OAA is a framework for integrating a community of heterogeneous software agents in a distributed environment. Psyclone [10] is a flexible middleware that can be used as a blackboard server for distributed, multi-module and multi-agent systems which may also be utilised.

3 Related Work

SmartKom [11] is a multimodal dialogue system currently being developed by a consortium of several academic and industrial partners. The system combines speech, gesture and facial expressions on the input and output side. The main scientific goal of SmartKom is to design new computational methods for the integration and mutual disambiguation of different modalities on a semantic and pragmatic level. SmartKom is a prototype system for a flexible multimodal human-machine interaction in two substantially different mobile environments, namely pedestrian and car. The system enables integrated trip planning using multimodal input and output. The key idea behind SmartKom is to develop a kernel system which can be used within several application scenarios. In a tourist navigation situation a user of SmartKom could ask a question about their friends who are using the same system. E.g. "Where are Tom and Lisa?", "What are they looking at?" SmartKom is developing an XML-based mark-up language called M3L (MultiModal Markup Language) for the semantic representation of all of the information that flows between the various processing components. SmartKom is similar to TeleMorph and TeleTuras in that it strives to provide a multimodal information service to the end-user. SmartKom-Mobile is specifically related to TeleTuras in the way it provides location sensitive information of interest to the user of a thin-client device about services or facilities in their vicinity. DEEP MAP [12, 13] is a prototype of a digital personal mobile tourist guide which integrates research from various areas of computer science: geo-information systems, data bases, natural language processing, intelligent user interfaces, knowledge representation, and more. The goal of Deep Map is to develop information technologies that can handle huge heterogeneous data collections, complex functionality and a variety of technologies, but are still accessible for untrained users. DEEP MAP is an intelligent information system that may assist the user in different situations and locations providing answers to queries such as- Where am I? How do I get from A to B? What attractions are near by? Where can I find a hotel/restaurant? How do I get to the nearest Italian restaurant? DEEP MAP displays a map which includes the user's current location and their destination, which are connected graphically by a line which follows the roads/streets interconnecting the two.

4 Conclusion

We have touched upon some aspects of Mobile Intelligent Multimedia Systems. Through an analysis of these systems a unique focus has been identified – "Bandwidth determined Mobile Multimodal Presentation". This paper has presented our proposed solution in the form of a Mobile Intelligent System called TeleMorph that dynamically morphs between output modalities depending on available network bandwidth. TeleMorph will be able to dynamically generate a multimedia presentation from semantic representations using output modalities that are determined by constraints that exist on a mobile device's wireless connection, the mobile device

itself and also those limitations experienced by the end user of the device. The output presentation will include Language and Vision modalities consisting of video, speech, non-speech audio and text. Input to the system will be in the form of speech, text and haptic deixis.

The objectives of TeleMorph are: (1) receive and interpret questions from the user, (2) map questions to multimodal semantic representation, (3) match multimodal representation to knowledge base to retrieve answer, (4) map answers to multimodal semantic representation, (5) monitor user preference or client side choice variations, (6) query bandwidth status, (7) detect client device constraints and limitations and (8) generate multimodal presentation based on constraint data. The architecture, data flow, and issues in the core modules of TeleMorph such as constraint determination and automatic modality selection are also given.

References

1. Holzman, T.G. (1999) Computer-human interface solutions for emergency medical care. *Interactions, 6(3),* 13-24.
2. Koch, U.O. (2000) Position-aware Speech-enabled Hand Held Tourist Information System. Semester 9 project report, Institute of Electronic Systems, Aalborg University, Denmark.
3. JCP (2002) Java Community Process. http://www.jcp.org/en/home/index
4. JSML & JSGF (2002). Java Community Process. http://www.jcp.org/en/home/index Site visited 30/09/2003.
5. Rutledge, L. (2001) SMIL 2.0: XML For Web Multimedia. In IEEE Internet Computing, Sept-Oct, 78-84.
6. McConnel, S. (1996) KTEXT and PC-PATR: Unification based tools for computer aided adaptation. In H. A. Black, A. Buseman, D. Payne and G. F. Simons (Eds.), Proceedings of the 1996 general CARLA conference, November 14-15, 39-95. Waxhaw, NC/Dallas: JAARS and Summer Institute of Linguistics.
7. Jensen, F.V. & Jianming, L. (1995) Hugin: a system for hypothesis driven data request. In Probabilistic Reasoning and Bayesian Belief Networks, A. Gammerman (ed.), 109-124, London, UK: Alfred Waller ltd.
8. Jeon, H., C. Petrie & M.R. Cutkosky (2000) JATLite: A Java Agent Infrastructure with Message Routing. IEEE Internet Computing Vol. 4, No. 2, Mar/Apr, 87-96.
9. Cheyer, A. & Martin, D. (2001) The Open Agent Architecture. Journal of Autonomous Agents and Multi-Agent Systems, Vol. 4, No. 1, March, 143-148.
10. Psyclone (2003) http://www.mindmakers.org/architectures.html
11. Wahlster, W.N. (2001) SmartKom A Transportable and Extensible Multimodal Dialogue System. International Seminar on Coordination and Fusion in MultiModal Interaction, Schloss Dagstuhl Int Conference and Research Center for Computer Science, Wadern, Saarland, Germany, 29 Oct-2 Nov.
12. Malaka, R. & A. Zipf (2000) DEEP MAP - Challenging IT Research in the Framework of a Tourist Information System. Proceedings of ENTER 2000, 7th International Congress on Tourism and Communications Technologies in Tourism, Barcelona (Spain), Springer Computer Science, Wien, NY.
13. Malaka, R. (2001) Multi-modal Interaction in Private Environments. International Seminar on Coordination and Fusion in MultiModal Interaction, Schloss Dagstuhl International Conference and Research Center for Computer Science, Wadern, Saarland, Germany, 29 October - 2 November.