

第六章

結論與未來研究方向

6.1 結論

在本論文中，首先針對虛擬環境系統、語音技術與 VoiceXML 相關議題加以探討，針對既有的 IMNet 虛擬環境提出互動式語音界面與 3D 虛擬環境整合的解決方案。接下來針對多人虛擬環境下的互動機制與問題分別加以說明、討論並提出解決方案，包含了對話鎖(DialogLock)、對話廣播(Broadcasting)、對話協議(Dialog Negotiate)與對話腳本擷取代理機制(Proxy-request)。

另外，對於 VoiceXML 對話模型在多人環境下所造成的問題也加以改良擴充，基於原有的 XAML 腳本語言，擴充成 XAML-V 對話腳本語言，並利用了物件導向的設計樣式(Design Patterns)與框架(Framework)等技術加以實作，對此設計的可行性加以驗證。在最後，我們提供了一個完整的範例，說明此一解決方案在多人虛擬環境中的運作方式。

在電腦硬體快速進步幫助之下，語音相關應用程式在個人電腦上的應用將會日漸普及，例如微軟在 Windows XP 作業系統中目前已內建了語音合成的服務。而隨著 3D 線上多人遊戲的盛行，多人虛擬環境技術也更形重要。基於語音是人類最常用的溝通方式的特性，我們相信語音與多人虛擬環境二者的整合將可以為虛擬環境使用者之間帶來更自然的互動。

6.2 未來研究方向

目前我們所設計的機制是使用 XAML 與 XAML-V 腳本語言來做到基本的動畫與語音同步控制。更加精緻的動畫與語音同步控制(如人臉動畫)必須基於 XAML 與 XAML-V 平台間傳送更精確的訊息。例如當 TTS 引擎將其語音佇列(Prompt Queue)完全播放完畢時，必須傳送一訊息給 XAML 平台，告知語音已播放完畢。

另外我們發現在虛擬環境中使用 VoiceXML 的對話模型的方式被廣為採用，因此很多 Dialog Script 都是以 VoiceXML 為基礎加以擴充或修改(包括 XAML-V)。但 VoiceXML 也有其先天的許多問題與限制，例如 VoiceXML 本身缺少和其它種類資源(如動畫)同步的機制與語法，也無法自由控制、或預期 TTS 播完的時間點，故未來 XAML-V 語法的延展可以朝這方面發展。

在效能方面，本研究中的 XAML-V 主要著重於在功能面的設計，並未針對傳送的效能或訊息數量做最佳化的考量。另外我們也認為在 Client 數量相當多的情況下 IMNet Server 可能成為整個平台效能的瓶頸，其中一種可能性是將 IMNet Server 置換為更有延展性的 JMS 訊息伺服器，以叢集(Cluster)方式達成效能的提升。

本系統中還有一些可以讓整個對話環境更加真實的改善空間。首先是在建立對話時，也尚未考慮二人的對話距離對於對話的影響，例如一旦人距離超過一定的長度，二人之前所建立的對話就必須被取消。建立對話前也應該先行檢查彼此的距離。最我們也希望未來能將 avatar 的 model 與 avatar 的語音加以客製化，可以讓使用者在登入前選擇並設定，如此一來虛擬環境中的每個 avatar 都能有不同的外觀及語音，進行對話時將更顯逼真。