Text Driven Talking Heads

Iain Brown

Undergraduate Dissertation
Computer Science
School of Informatics
University of Edinburgh

2015

Abstract

The aim of this project is to build a head motion synthesizer for a lifelike animated avatar. The head motions will be predicted entirely from the text of transcribed speech with the aim of finding a mapping between the text and natural head motions. Unlike previous areas of research where the head motions are generated from recorded speech.

Table of Contents

Chapter 1

Introduction

The aim of this project is to find a mapping between spoken text and discrete head motions in order to synthesise realistic, natural head motions for a life like avatar. Currently we have a system that generates head motions using Hidden Markov Models and recorded speech, but this project will focus only on the text.

1.1 Text to Speech

Speech contains a lot of information to help portray what the person is trying to say to the listener. Speech has stresses, intonation and pauses which all help to further our understanding of what is being said. When we read transcribed text this information is simply just not there. In order for us to get as much data out of transcribed text we need to synthesise spoken speech using the transcribed text so we can generate information that will help us synthesise realistic head motions.

To do this we will use a text to speech synthesis system called Festival. Festival uses many different Natural Language Processing techniques in the text to speech pipeline to break down the sentence into an Ütterance¨. An Utterance will hold all the information about the synthesised speech from the text. Festival is a great research tool in text to speech because each step in the pipeline can be individually inspected so we can understand what is happening and each stage and how this will affect the generated speech.

1.2 Data Set

The data used for this project consists of audio recordings with motion tracked data that was recorded from many different speakers. The recordings are accompanied with transcriptions of the speech. As motion can be represented with many different parameters we will convert the recorded motions into into Euler angles which represent the head motions in terms of three values : yaw, pitch and roll. This is suitable for the project as we are focusing on the head motions alone and translation will not be considered.

1.3 3D Modelling

To visualise the head motions that we will be synthesising we needed animation software that we would allow us to manipulate characters autonomously with the aid of scripts. The first software I looked at was called Poser developed by SmithMicro. The software widely used by animators and researchers in this area of Computer Science and the software focuses on 3D character animation allowing the user to load in preset character models from a library. Poser has a python extension called PoserPython with a comprehensive online manual and direct integration with the software out of the box so that character models can be manipulated using scripts. Poser can be loaded with many different extensions to streamline animation and there is an extension that allows character models to automatically lip-sync with audio and as the project aims to create realistic 3D avatars this feature may prove useful in the evaluation stage.

Another piece of software I looked into was an open source animation software called Blender. Like Poser this software has a python extension called Blender/Python allowing the manipulation of objects. The software itself however has a steeper learning curve than Poser so I chose to use Poser.

Chapter 2

Background

2.1 Festival

Festival has many different steps in the Text to Speech pipeline. Tokenisation breaks the string of characters into a list of tokens, this removes whitespace and identifies the words in the sentence which will then be assigned Parts-of-speech tags. Parts-of-Speech tags (POS tags) indicate what syntactic classification the word is, meaning that the word will be assigned a value indicating whether it is a determiner, noun, verb, etc. Assigning POS tags is a very important step in the text to speech pipeline as this helps understand the structure of the text in order to perform phrase break predication, in which Festival will assign a phrase break strength after each word indicating how likely this is the end of a phrase and the next word will be the beginning of a phrase. This plays a large role when we actually synthesise speech as Festival will be able to calculate where the stress in the speech will be, which will affect the head motions relating to the speech.

Using all the information calculated so far Festival can then break the utterance into speech segments and perform a lexical lookup to find the pronunciation of said segments. Festival uses these pronunciations, represented by a phonetic spelling to assign intonations to the segments called ToBI (Tones and break indices) tags. ToBi tags are represented symbolically using a combination of letters and symbols. Durations can be calculated using a myriad of methods in Festival, the default method of assigning the duration is simply just a constant value, however this can be swapped out in place of more interesting methods like Klatt Durations or CART Durations.

Once the durations and intonations have been assigned Festival then applies signal processing methods in order to perform waveform generation for the utterance to output the resulting speech. The data contained in the utterance will be vital in predicting head motions and Festival was the obvious choice for a Text to Speech system as the components can be examined individually.

2.2 Hidden Markov Models and HTK

To predict head motions using Machine learning techniques we will be using a toolkit for handling Hidden Markov Models called HTK. Hidden Markov Models are finite state machines with probabilities assigned to their transitions and are used widely in machine learning for pattern matching. Their power for machine learning comes from it’s memoryless Markov property stating that the future is independent of the past, meaning at

Chapter 3

Text-Driven Head Motion System

3.1 Discrete head motion mappings

In order to build the Text-Driven Head Motion System we will be mapping discrete head motions to the text. This will be done in two ways : A Rule based method that will be manually coded and a machine learning based approach which will use a portion of the dataset as training data to predict head motions automatically using Hidden Markov Models. I will represent the discrete head motions and their relation to the transcribed speech as a Finite State machine with the states representing the words in the speech (or phonemes to give more precise control) and transitions being the discrete head motions between these states.

This means that any transition between states will be a combination of head motions which will sum together to get the overall shape of text. An example of this would be a nodding motion being assigned to the whole text with a slight tilt being assigned to a certain state within that.

We can use the trigonometric sine function to generate smooth head motions which will be assigned to the discrete head motions rather than use example from the recorded data, as a ”nod” in the recordings may have additional data unrelated to the nod and so would be a bad baseline, potentially causing issues with the final synthesised head motions.

3.2 Rule Based Method

To be implemented.

- CART Trees
- Sentiment Analysis

3.3 Machine Learning Based Method

To be implemented.

- HMM's
- Hidden Markov Toolkit

3.4 Evaluation

To evaluate the Text-Driven Head Motion System I will be performing both objective and subjective evaluation. This is to ensure that the Text Driven Talking Head system renders synthesises head motions that seem natural to humans and to evaluate how accurate the system is at synthesising head motions that match the original recorded head motions.

To evaluate the head motions system objectively I will use a local distance measure periodically in the animation to check how close the synthesised head position is to the recorded. This may however provide bad results if the head motions are of the same type (yaw, pitch and roll) but different directions. For example, if the recorded head tilted to the right and the synthesised head motions tilted to the left. Another way to perform objective analysis could be to convert the recorded head motions into their discrete head motion counterparts and compare the discrete head motions directly. This method would leave some area for variability and give good results if the head motions are of the same type but may be difficult to convert the recorded head motions into discrete head motions.

To set up the subjective analysis environment, we will show recordings of the the real-life recorded head motions with recordings of the synthesised head motions side-by-side to test subjects and ask them to evaluate the synthesised head motions. In order to ensure that the test subject is only evaluating the head motions and is not influenced by other factors we will try to keep the environment as similar as possible. This will involve mapping the recorded head motions onto the same 3D character model as used for the synthesised head motions and using the synthesised audio for the transcribed speech for both heads. In the case that the audio recording and the synthesised speech do not map directly we will perform dynamic time warping to fit the recorded head motions with the synthesised speech.

Chapter 4

Results / Progress

Completed

- Research into festival
- Research into HTK
- Built very basic system to tie together Festival and Poser

Incomplete/ Yet to be implemented

- Rule-based Method
- Machine Learning Based Method