CircumReality text-to-speech, a talking speech recognizer

Mike Rozak

μXac, Darwin, NT, Australia

Mike@mXac.com.au, http://www.CircumReality.com

Abstract

The CircumReality text-to-speech engine's mean opinion (MOS) and similarity-to-original scores have improved significantly over the last three Blizzard Challenges [1] [2]. MOS has increased from 1.3 in 2007 to 2.8 in 2009. This paper describes the algorithmic improvements made to the CircumReality engine between the 2008 and 2009 Blizzard Challenges. The most significant improvements stemmed from a shift in the underlying philosophy of the engine: integrating automatic speech recognition (ASR) into the speech synthesis engine and creating a "talking speech recognizer".

Index Terms: speech synthesis, speech recognition, games, unit selection, Blizzard Challenge

1. Introduction

The annual Blizzard Challenge allows text-to-speech researchers to compare their engine technologies against one another by providing a common speech database from which all entered voices are created [3]. Such a test allows speech researchers to eliminate voice-database quality, and to a lesser extent, hand-tuning, as factors in synthesized voice quality. Researchers employ the test's results, including associated Blizzard-Challenge papers explaining other entrants' results, to improve their engines.

The CircumReality text-to-speech engine has improved significantly over the three years that it has been entered in the Blizzard Challenge. (See figure 1.)

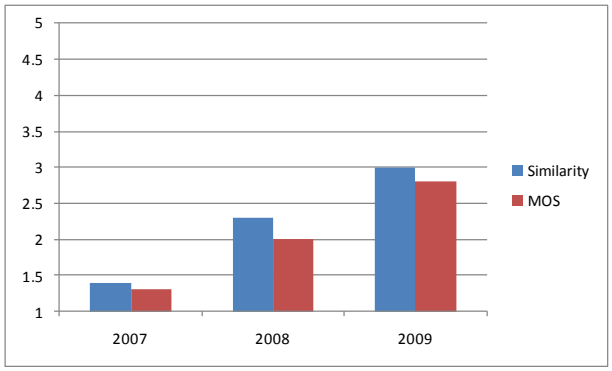

Figure 1: CircumReality text-to-speech engine similarity and mean-opinion scores (MOS) over three Blizzard Challenges.

The improvements to CircumReality's speech quality came from two directions:

- **Acoustic features** – The use of TD-PSOLA proved superior to time-domain stretched PCM (in the 2008 entry [2]) or additive sine-wave synthesis (in the 2007 entry [1]).
- **Using ASR for target and join costs** – A new philosophical foundation for the text-to-speech engine was employed: The process of speech synthesis was understood in terms of a "talking speech recognizer", and ASR (automatic speech recognition) was tightly incorporated into the synthesis process.

2. Blizzard Challenge 2009 results compared to 2008

CircumReality was entered in the 2009 Blizzard Challenge as voice "H". The CircumReality engine was entered for the EH1 (10 hours of speech data), EH2 (1 hour of Arctic speech data) and ES1 (100 sentences from the Arctic dataset) tests. A Mandarin Chinese voice was generated, but not entered due to the CircumReality engine's poor synthesis of Chinese.

The CircumReality engine uses unit-selection synthesis; unlike standard unit-selection synthesizers [4, pp 475-493], CircumReality runs its own speech recognizer in tandem with voice generation and synthesis [2] to help determine which units to select. Voice creation is mostly automated, although for the Roger dataset, some sentence utterance groups were manually eliminated from the prosody model because they contained atypical prosody.

In 2008, the CircumReality engine ranked at the bottom of the submitted entries [2]. The engine performed significantly better in the 2009 Blizzard Challenge, achieving better-than-average MOS and similarity scores. (See figure 2.) Interestingly, the engine didn't do well on the word-error-rate test, as will be discussed later.

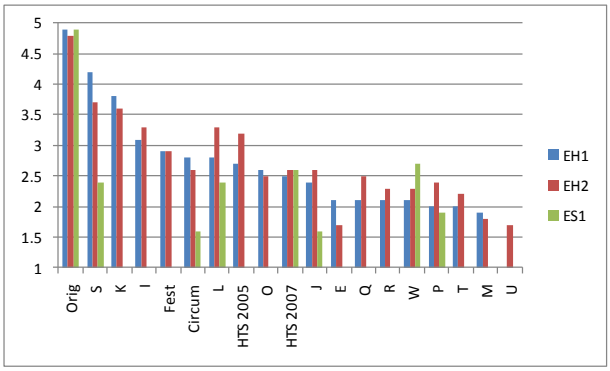

Figure 2: Mean opinion score (MOS) for EH1, EH2, and ES1. "Orig" is the original voice, "Circum" is CircumReality. "Fest", "HTS 2007", and "HTS 2005" are reference engines.

3. A talking speech recognizer

Stereotypical unit-selection synthesizers employ ASR for phoneme segmentation [4, pp 467-471], and may use the ASR phoneme scores, along with extremes in F0, energy, and duration, to eliminate the "worst" sounding phonemes.

From early in its development, the CircumReality text-to-speech engine has incorporated units' ASR scores in the unit-select Viterbi search. [1]

CircumReality's 2008 engine (CR2008) used ASR extensively. [2] Most notably, the target cost for F0 and duration mismatches were calculated using ASR; target costs no longer needed to be manually generated. Target costs for mismatched start/end of a word, and mismatched left/right contexts were also calculated using ASR. Unit join costs were calculated using the same feature-comparison algorithms employed by ASR.

CircumReality's 2009 entry (CR2009) integrated ASR even more extensively than CR2008. ASR is so intertwined into the text-to-speech algorithms that CircumReality's synthesis and ASR are inseparable.

All values used in the unit-selection Viterbi search come directly from, or are derived from ASR. Many values in the prosody model also originate from ASR.

In other words: CircumReality's built-in ASR "listens to" and fine-tunes the output of the text-to-speech algorithms before they're heard by the listener.

Significant engine changes between CR2008 and CR2009 that affected the 2009 Blizzard Challenge are:

- **Phoneme categories** – CR2008 grouped phonemes into four categories when using ASR to calculate target costs: voiced plosive, unvoiced plosive, voiced non-plosive, and unvoiced non-plosive. [2] CR2009 increased the number of categories to sixteen, resulting in more accurate target cost calculations.
- **Half-phone target cost base** – CR2008 and CR2009 synthesize using half units. CR2008 incorporated the unit's context-dependent ASR score as the base value for the unit-selection score, using the same value for both the left and right halves of the unit. CR2009 uses ASR to calculate unique values for the left and right halves of the unit.
- **Explicitly calculated left/right context mismatches** – In CR2008, if a unit with mismatched left/right phoneme context was used, a score penalty (calculated using ASR) would be applied. CR2009 still does this, in most cases. However, CR2009 identifies units with mismatched left/right contexts that are likely substitutes, and then explicitly calculates the unit-substitution's score against the ASR context-dependent phoneme model of the desired unit. Doing so eliminates the need to use the estimated mismatch score penalty, producing a more-accurate target cost for likely substitutions.
- **Join cost calculation using a triangular window** – In CR2008, join costs were calculated using an impulse window, using ASR to compare frames to the immediate left and right of the join. [2] CR2009 uses a triangular window with a width of around half a phoneme.
- **Target and estimated join costs calculated per voice** – CR2008's target and estimated join costs were calculated from 10 hours of recordings of my own voice. CR2009 calculates the target and estimated join costs from the Roger voice, improving target cost accuracy when synthesizing the Roger voice. The resulting target-cost values are significantly different to those derived from my own voice.
- **TD-PSOLA target costs** – It is well known that TD-PSOLA distorts the original signal, and "as a rule of thumb" [4, pp 416] can only be used to double or halve the duration of a unit, and increase or decrease its F0 by half an octave. CR2009 used ASR to determine how much changing the duration and/or F0 using TD-PSOLA affected the speech quality, and included this into the target cost. I won't detail the CR2009 algorithm to derive TD-PSOLA costs because recently-improved algorithms (see below) have made the TD-PSOLA target-cost algorithm employed in CR2009 obsolete.
- **"Snap to" F0 and duration affected by target costs** – Stereotypical unit-selection synthesizers maintain the original units' F0 and duration. Due to transplanted prosody requirements for games, the CircumReality engine modifies F0 and duration using TD-PSOLA. To minimize the signal distortions created by TD-PSOLA, CR2008 and CR2009 adjust the F0 and duration of a unit away from the values requested by the prosody model, and towards the F0 and duration of the original unit. This approach reduces prosody quality to improve acoustic quality. In CR2009, the amount of adjustment is controlled by the target-cost penalty per octave shift of F0 or doubling of duration. For example: Phoneme groups that have higher F0 target-cost penalties, meaning that they don't sound as good when pitch shifted using TD-PSOLA, have their F0 weighted more towards the unit's original F0.
- **TD-PSOLA** – CR2008 synthesized audio using time-domain stretched PCM, causing audible artifacts when F0 was modified even slightly. [2] CR2009 used TD-PSOLA, resulting in improved acoustic synthesis.
- **Prosody model** – The ASR-calculated F0, duration, and energy target costs are employed by the prosody model to estimate how perceptible altering a syllable's F0, duration, or energy is. Such values are only a guestimate, to be used until better approaches for calculating prosody-specific F0, duration, and energy target costs can be devised.

Other significant improvements to CR2009 didn't affect the 2009 Blizzard Challenge:

- **Small voices** – The CR2008 and CR2009 Blizzard-Challenge voices used around 350,000 units and several gigabytes on disk. Text-to-speech voices for games must be smaller, around 8000 units and 30 megabytes. CR2008 eliminated units by retaining the 8000 best-scoring, most-commonly-used phoneme sequences, based on an average of the unit's ASR-generated base score. CR2009 includes estimated join costs for non-contiguous units between the first two and last two phonemes of the sequences, also calculated by ASR. For example: Estimated join costs around the phoneme "t" are always high, due to coarticulation. Conversely, "m" has low estimated join costs. As a result, triphone sequences with "t" occurring in the middle of sequence are more likely to be included in the 8000-unit voice than triphone sequences with an "m" in the middle.
- **Randomly generate several sentences and select the "best sounding" one** – CR2009 can randomly generate several different prosodies for a given sentence, along with randomly selected alternative pronunciations from the lexicon. All variations are synthesized, and the synthesized sentence with the best unit-selection score is spoken. This technique was not used for the Blizzard

Challenge 2009 because it proved to be too slow, and produce only a marginal improvement in speech quality.

- **TD-PSOLA target cost improvements** – After submitting the synthesized results for Blizzard Challenge 2009, further improvements were made to calculating TD-PSOLA target costs. The CircumReality engine now uses pitch-detection confidence scores from the training data to adjust the "per octave shift" and "per duration doubling" target costs of TD-PSOLA. In general, the higher the pitch-detect confidence, the lower the TD-PSOLA target cost. ASR is used to automatically calculate the TD-PSOLA per-octave/duration target costs as a function of pitch-detection confidence.

4. Conclusions and future work

TD-PSOLA and the "talking speech recognizer" philosophy significantly improved CircumReality's MOS and similarity scores.

However, CircumReality's 2009 "word error rate" was still very high. (See figure 3.)

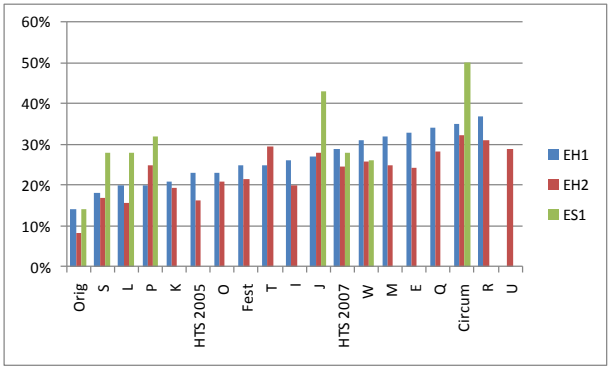

Figure 3: Word error rate

The high word-error rate is unexpected, and needs to be explored. Several possible causes for the word-error rate will be investigated:

- **The ASR algorithm may not be accurate enough** – CircumReality's ASR algorithms haven't yet been tested for accuracy. Future plans involve writing a phoneme-based ASR accuracy test, and then fine-tuning constants and algorithms to improve ASR accuracy.
- **Join cost triangle window size** – Join costs are calculated by comparing the boundaries of a join using a triangle window of approximately half a phoneme. Since the units of the beam search are "frame comparison error * time", a shorter-duration window causes the join cost to affect the beam search more. Thus, a shorter-duration triangle window encourages more non-contiguous units. The word-error-rate listening test was based on short, confusable word pairs, implying that a longer triangle-window would encourage contiguous units and reduce the word error rate.
- **Join cost vs. target cost weight** – In CR2009, the join cost score is combined with the target cost score using a weight of 1.0. The reasoning for using 1.0 may need to be re-examined; a different weight might make more logical sense. Increasing the join cost weight would encourage contiguous units.

The CircumReality text-to-speech engine was created for the CircumReality game [5], and all work on the engine is done with game development in mind. The 2009 Blizzard Challenge has provided some information relevant to game development:

- **Minimizing voice-recording costs** – While 10,000 sentences for each voice would be ideal, recording so many sentences isn't possible on a small financial budget. 1000 sentences appear to be the minimum number of recordings needed before MOS declines dramatically. (See figure 2.) CircumReality's low MOS for ES1 (generated from 100 sentences) illustrates the rapid drop-off in quality resulting from less data. Due to the CircumReality game's low budget, most voice data will come from free public sources where 1-hour voiced databases are common, but 10-hour voice databases are rare.
- **Quality vs. quantity –** Another voice-design tradeoff is whether the game should ship with a couple of large voices generated from 10 hours of speech, and then use extensive voice transformations to create voices for one hundred characters, or to ship with 20-40 smaller voices and employ only minor voice transformation to cover the one hundred characters. HMM synthesis using highly-parameterized speech audio would enable both significant voice transformations and small voices, with HTS 2007's ES1 matching its EH1 and EH2 scores. (See figure 2.) But, from the Blizzard Challenge 2008's overall results, it is obvious that "the best" concatenative PSOLA synthesizers still have a significantly higher MOS for EH2 (small voices) than "the best" parameterized-speech HMM synthesizers have for EH1. These results show that 20-40 smaller PSOLA voices will produce a better over-all MOS than highly-parameterized voices.
- **Prosody** – Listening to the restaurant query-responses sub-test of the 2009 Blizzard Challenge clearly demonstrated how poor CR2009's prosody was. Unfortunately, separate test results weren't provided, so no numerical comparison is possible; I suspect CircumReality's MOS would be relatively higher (compared to other entrants) if the restaurant-query test results were removed. However, better prosody is not that critical for games. Long sentences such as those used in the restaurant-query sub-test don't appear often in games; players get bored listening to even medium-length sentences. Furthermore, half of the sentences that are spoken during gameplay can be "prerecorded" with transplanted prosody, overriding the lower-quality synthesized prosody.

- **Expert speech listener bias** – "Speech experts" consistently gave all entrants the same or higher MOS and similarity scores. In terms of gameplay, this implies that players will "grow accustomed to" text-to-speech voices over time. (See figure 4.)

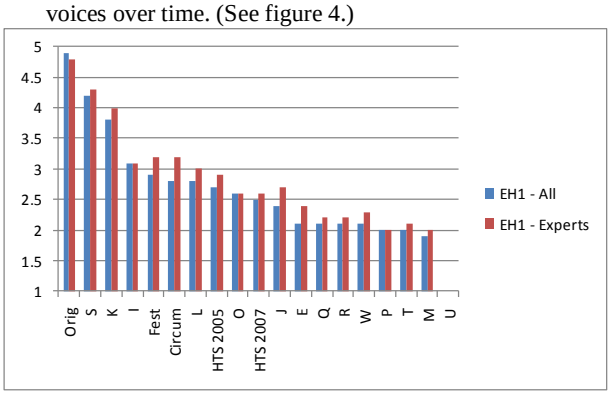

Figure 4: Expert speech listener bias

5. References

[1] Rozak, M., "Text-to-speech Designed for a Massively Multiplayer Online Role-Playing Game (MMORPG)", in *The Blizzard Challenge 2007*, Bonn, Germany. mXac. Online: http://festvox.org/blizzard/bc2007/index.html, accessed on 19 July 2009.

[2] Rozak, M., "CircumReality functionality delta: Blizzard Challenge 2007 to 2008", in *The Blizzard Challenge 2008*, Brisbane, Australia. mXac. Online: http://festvox.org/blizzard/blizzard2008.html, accessed on 19 July 2009.

[3] Karaiskos, V., King, S., Clark, R., Mayo, C., "The Blizzard Challenge 2008", in *The Blizzard Challenge 2008*, Brisbane, Australia. University of Edinburgh. Online: http://festvox.org/blizzard/blizzard2008.html, accessed on 19 July 2009.

[4] Taylor, P., *Text-to-Speech Synthesis*, 2009, New York, Cambridge University Press.

[5] Rozak, M., "What is CircumReality?", mXac. Online: http://www.CircumReality.com, accessed on 19 July 2009.