社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

HMM 音声合成システム (HTS) の開発

全　炳河[†]　大浦　圭一郎[†]　能勢　隆[††]　山岸　順一[†††]　酒向　慎司[†]
戸田　智基[††††]　益子　貴史[††]　ブラック　アラン[†††††]　徳田　恵一[†]

† 名古屋工業大学 情報工学専攻　〒 466–8555 愛知県名古屋市昭和区御器所町
†† 東京工業大学 物理情報システム専攻　〒 226–8502 神奈川県横浜市緑区長津田町 4259–G2–4
††† エジンバラ大学 音声技術研究所
†††† 奈良先端科学技術大学院大学　〒 630–0192 奈良県生駒市高山町 8916–5
††††† カーネギーメロン大学 言語技術研究所
E-mail: †{zen,uratec,sako,tokuda}@sp.nitech.ac.jp, ††takashi.nose@ip.titech.ac.jp, †††jyamagis@inf.ed.ac.uk,
††††tomoki@is.naist.jp, ††††††awb@cs.cmu.edu

あらまし　近年，隠れマルコフモデル (HMM) に基づく統計的パラメトリック音声合成方式が注目されている．本方式では，音声スペクトル・励振源・継続長がコンテキスト依存 HMM により同時にモデル化される．音声合成時は，合成したい文章に対応する HMM からの出力確率が最大となるよう，継続長・スペクトル・励振源系列を決定した後，音声合成フィルタを用いて波形が出力される．2002 年より我々は，HMM に基づく音声合成のための研究・開発ツール「HMM 音声合成システム (HTS)」を，オープンソースソフトウェアとして公開してきた．本報告では，その最新の開発状況と今後の予定について述べる．

キーワード　音声情報処理，音声合成，隠れマルコフモデル

Recent developments of the HMM-based speech synthesis system (HTS)

Heiga ZEN[†], Keiichiro OURA[†], Takashi NOSE[††], Junichi YAMAGISHI[†††], Shinji SAKO[†], Tomoki TODA[††††], Takashi MASUKO[††], Alan W. BLACK[†††††], and Keiichi TOKUDA[†]

† Department of Computer Science and Engineering, Nagoya Institute of Technology
†† Graduate School of Information Science, Nara Institute of Science and Technology
††† The Centre for Speech Technology Research, University of Edinburgh
2 Buccleuch Place, Edinburgh, EH8 9LW, United Kingdom
†††† Graduate School of Information Science, Nara Institute of Science and Technology
††††† Language Technology Institute, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213, USA
E-mail: †{zen,uratec,sako,tokuda}@sp.nitech.ac.jp, ††takashi.nose@ip.titech.ac.jp, †††jyamagis@inf.ed.ac.uk,
††††tomoki@is.naist.jp, ††††††awb@cs.cmu.edu

Abstract　A statistical parametric speech synthesis approach based on hidden Markov models (HMMs) has grown in popularity over the last few years. In this approach, spectrum, excitation, and duration of speech are simultaneously modeled by context-dependent HMMs, and speech waveforms are generated from the HMMs themselves. Since December 2002, we have publicly released an open-source software toolkit named "HMM-based speech synthesis system (HTS)" to provide a research and development toolkit of statistical parametric speech synthesis. This paper describes recent developments of HTS in detail, as well as future release plans.

Key words　Speech processing, speech synthesis, hidden Markov model

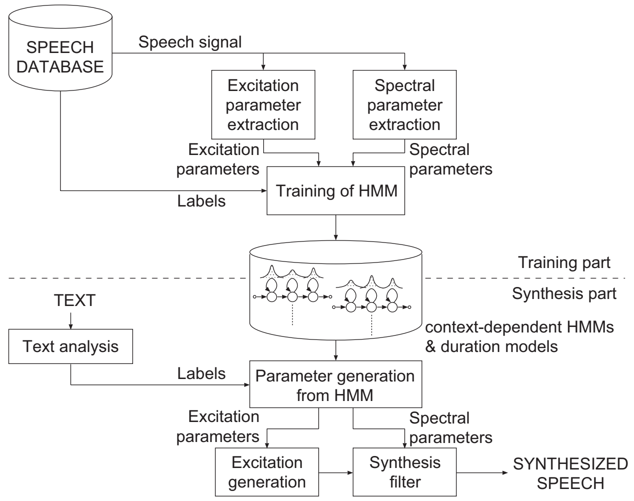

図 1 Overview of HMM-based speech synthesis.

1. Introduction

A statistical parametric speech synthesis approach based on hidden Markov models (HMMs) has grown in popularity over the last few years [1]. In this approach, context-dependent HMMs are estimated from databases of natural speech, and speech waveforms are generated from the HMMs themselves. This framework makes it possible to model different speaking-styles without recording large speech databases. For example, adaptation [2], interpolation [3], and eigenvoice techniques [4] were applied to this system, and it was found that voice characteristics could be modified.

Since December 2002, we have publicly released an open-source software toolkit named "HMM-based speech synthesis system (HTS)" [5] to provide a research and development platform of statistical parametric speech synthesis. Currently various organizations use it to conduct their own research projects, and we believe that it has contributed significantly to the success of HMM-based speech synthesis today. This report describes the recent developments of this system as well as the future release plans.

The rest of this report is organized as follows: Section 2 reviews statistical parametric speech synthesis. In Section 3, details of HTS are described. Other applications of HTS are presented in Section 4. Concluding remarks and future release plans are presented in the final section.

2. Statistical parametric speech synthesis

Text-to-speech synthesis can be viewed as an inverse procedure of speech recognition. The goal of any text-to-speech synthesizer is to take a word sequence $\boldsymbol{w} = \{w_1, \ldots, w_N\}$ as its input and produce an acoustic speech waveform $\boldsymbol{o} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$. In a typical system, contextual factors such as accent, lexical stress, part-of-speech, and phrase boundary are assigned to a given word sequence $\boldsymbol{w}$ by a natural language processing engine, and then $\boldsymbol{w}$ is mapped into the corresponding context-dependent sub-word sequence $\boldsymbol{u} = \{u_1, \ldots, u_M\}$. Finally, a speech waveform $\boldsymbol{o}$ is synthesized for $\boldsymbol{u}$.

Most of state-of-the-art speech synthesis systems are based on large amount of speech data. This type of approach is generally called as corpus-based speech synthesis. This approach makes it possible to dramatically improve the naturalness of synthesized speech compared with the early speech synthesis systems.

One of the major approaches in corpus-based speech synthesis is sample-based one, such as unit-selection [6]. In this approach, speech data are segmented into small units, i.e. HMM state, half-phone, phone, diphone, or syllable, and stored. Then a unit sequence corresponding to a given context-dependent sub-word sequence is selected by minimizing its total cost consisted of target and concatenation costs [6]. These cost functions have been formed from a variety of heuristic or ad hoc quality measures based on features of the acoustic signal and given texts. Recently, target and concatenation cost functions based on statistical models have been proposed and investigated [7–11].

Another major approach is statistical parametric one, such as HMM-based speech synthesis [1]. It generates a speech parameter vector sequence $\boldsymbol{o} = \{\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T\}$ with maximum a posteriori (MAP) probability given the context-dependent sub-word sequence $\boldsymbol{u}$ as follows:

$$\hat{\boldsymbol{o}} = \arg\max_{\boldsymbol{o}} P(\boldsymbol{o} \mid \boldsymbol{u}). \tag{1}$$

Although any kind of generative models can be applied to represent $P(\boldsymbol{o} \mid \boldsymbol{u})$, currently HMMs are widely used.

Figure 1 overviews HMM-based speech synthesis. It consists of training and synthesis parts. The training part is similar to that used in speech recognition. The main difference is that both spectrum (e.g., mel-cepstral coefficients and their dynamic features) and excitation (e.g., $\log F_0$ and its dynamic features) parameters are extracted from a speech database and modeled by context-dependent HMMs (phonetic, linguistic, and prosodic contexts are taken into account). To model variable dimensional parameter sequences such as $\log F_0$ with unvoiced regions, multi-space probability distributions (MSD) [12] are used for state output probability density functions (PDFs). Each HMM has its state duration PDFs to capture the temporal structure of speech. As a result, spectrum, excitation, and durations are modeled simultaneously in a unified HMM framework [1]. The synthesis part does the inverse operation of speech recognition. First, an arbitrarily given text to be synthesized is converted to a context-dependent label sequence, and then a sentence HMM is constructed by concatenating the context-dependent HMMs according to the label sequence. Second, state durations of the sentence HMM are determined based on the state duration PDFs. Third, the speech parameter generation algorithm generates sequences of spectral and excitation parameters that maximize their output probabilities under the constraints between static and dynamic features [13]. Finally, a speech waveform is synthesized directly from the generated spectral and excitation parameters using a speech synthesis filter. The most attractive part of this system is that its voice characteristics, speaking styles, or emotions can easily be modified by transforming HMM parameters using various techniques, such as adaptation [2], interpolation [14], or eigenvoice [4].

3. HTS: A software toolkit for HMM-based speech synthesis

3.1 Outline

The HMM-based speech synthesis system (HTS) has been being developed by the HTS working group as an extension of the HMM toolkit (HTK) [15]. The history of the main modifications which we have made are listed below:

- **Version 1.0 (December 2002)**
  - Tree-based clustering based on the MDL criterion [16].
  - Stream-dependent tree-based clustering [1].
  - Multi-space probability distributions (MSD) [12].
  - State duration modeling and clustering [17].
  - Speech parameter generation algorithm [13].
  - Demo using the CMU Communicator database.
- **Version 1.1 (May 2003)**
  - Small run-time synthesis engine.
  - Demo using the CSTR TIMIT database.
  - HTS voices for the Festival speech synthesis system [18].
- **Version 1.1.1 (December 2003)**
  - Variance flooring for MSD-HMMs.
  - Post-filtering [19].
  - Demo using the CMU ARCTIC database.
  - Demo using the Nitech Japanese database.
  - HTS voice for the Galatea toolkit [20].

The source code of HTS is released as a patch for HTK. Although the patch is released under a free software license similar to the MIT license, once the patch is applied users must obey the license of HTK.[注1]

3.2 HTS version 2.0 / 2.0.1

After an interval of three years, HTS version 2.0 was released in December 2006. This was a major update and included a number of new features and fixes:

- Terms about redistributions in binary form were added to the HTS license.
- `HCompV` (global mean and variance calculation tool) accumulates statistics in double precision. For large databases the previous versions often suffered from numerical errors.
- `HRest` (Baum-Welch re-estimation tool for a single HMM) can generate state duration PDFs [17] with the `-g` option.
- Phoneme boundaries can be given to `HERest` (embedded Baum-Welch re-estimation tool) using the `-e` option. This can reduce computational cost and improve phoneme segmentation accuracy [21]. We may also specify subset of boundaries (e.g, pause positions).
- Reduced-memory implementation of tree-based clustering in `HHEd` (a tool for manipulating HMM definitions) with the `-r` option. For large databases the previous versions sometimes consumed huge memory.
- Each decision tree can have a name with regular expressions (`HHEd` with the `-p` option). As a result, two different trees can be constructed for consonants and vowels respectively.
- Flexible model structures in `HMGenS` (speech parameter generation tool). In the previous versions, we assumed that the first HMM stream is mel-cepstral coefficients and the others are for log F_0. Now we can specify model structures using the configuration variables `PDFSTRSIZE` and `PDFSTRORDER`. Non-left-to-right model topologies (e.g., ergodic HMM), Gaussian mixtures, and full covariance matrices are also supported.
- Speech parameter generation algorithm based on the expectation-maximization (EM) algorithm (the Case 3 algorithm in [13]) in `HMGenS`. Users can select generation algorithms using the `-c` option.
- Random generation algorithm [22] in `HMGenS`. Users can turn on this function by setting a configuration variable `RNDPG=TRUE`.
- State or phoneme-level alignments can be given to `HMGenS`.
- The interface of `HMGenS` has been switched from `HHEd`-style to `HERest`-style.
- Various kinds of linear transformations for MSD-HMMs are supported in `HERest`.
  - Constrained and unconstrained maximum likelihood linear regression (MLLR) based adaptation [23].
  - Adaptive training based on constrained MLLR [23].
  - Precision matrix modeling based on semi-tied covariance matrices [24].
  - Heteroscedastic linear discriminant analysis (HLDA) based feature transform [25].
  - Phonetic decision trees can be used to define regression classes for adaptation [26]
  - Adapted HMMs can be converted to the run-time synthesis engine format.
- Maximum a posteriori (MAP) adaptation [27] for MSD-HMMs in `HERest`.

HTS version 2.0.1 was a bug fix version. The new features in this version are as follows:

- Band structure for linear transforms [28].
- Speaker interpolation [3].
- Stream-dependent variance flooring scales.
- Demo scripts support LSP-type spectral parameters.
- β version of the runtime synthesis engine API.

3.3 New features in version 2.1

We are planning to release HTS version 2.1 at the end of March 2008. This version will include four important new features: hidden semi-Markov models (HSMMs) [29,30], the speech parameter generation algorithm considering global variance (GV) [31], advanced adaptation techniques [32], and stable version of runtime synthesis engine API.

Note that HTS version 2.1, with the STRAIGHT analysis/synthesis technique [33], provides the ability to construct the state-of-the-art HMM-based speech synthesis systems developed for the past Blizzard Challenge events [34–36].

3.3.1 Hidden semi-Markov model

In HMM-based speech synthesis, rhythm and tempo are controlled by state duration PDFs. They are estimated from statistical variables obtained at the last iteration of the forward-backward algorithm, and then clustered by a decision tree-based context-clustering algorithm: they are not re-estimated in the Baum-Welch iteration [17]. In the synthesis stage, we construct a sentence HMM and determine its state durations so as to maximize their probabilities. Then, speech parameter vector sequences are generated. However, there is an inconsistency: although parameters of HMMs are estimated without explicit state duration PDFs, speech parameter vector

[注1]：The HTK license prohibits redistribution and commercial use.

sequences are generated from HMMs using the explicit state duration PDFs. This inconsistency can degrade the quality of synthesized speech.

To resolve the discrepancy, HSMMs [37], which can be viewed as HMMs with explicit state duration PDFs, were introduced into the training part [29]. The use of HSMMs makes it possible to simultaneously re-estimate state output and duration PDFs. The adaptation and adaptive training techniques for HSMMs were also derived [30]. Zen et al. reported small improvements in speaker-dependent systems [29]. However, Tachibana et al. reported that the use of HSMM was essential to adapt state durations PDFs [38]. The HSMM was also successfully applied to speech recognition [39].

3.3.2 Speech parameter generation algorithm considering global variance

In the basic system, the speech parameter generation algorithm is used to generate spectral and excitation parameters from the HMMs [13]. By taking account of constraints between the static and dynamic features, it can generate smooth speech parameter trajectories. However, the generated spectral and excitation parameters are often excessively smooth compared with the natural speech. We expect that the statistical modeling process removes the details of spectral structure. Although this smoothing surely causes error reduction of the spectral generation, it also causes the degradation of naturalness of synthesized speech because those removed structures are still necessary to synthesize high-quality speech. To suppress this problem, Toda et al. proposed a speech parameter generation algorithm considering global variance (GV) [31].

This algorithm iteratively maximizes the following objective function with respect to a speech parameter vector sequence $\boldsymbol{c} = \left[\boldsymbol{c}_1^\top, \dots, \boldsymbol{c}_T^\top\right]^\top$ (static features only):

$$\mathcal{L}(\boldsymbol{c}) = w \log P(\boldsymbol{W}\boldsymbol{c} \mid \boldsymbol{q}, \lambda) + \log P(v(\boldsymbol{c}) \mid \lambda_v) \tag{2}$$

where λ is a sentence HSMM, $\boldsymbol{q} = \{q_1, \dots, q_T\}$ is a state sequence determined by state duration PDFs, $\boldsymbol{W}$ is a window matrix which appends delta and delta-delta features to $\boldsymbol{c}$, w is a weight for the state output probability, $v(\boldsymbol{c})$ is a GV of $\boldsymbol{c}$ which is defined as an intra-utterance variance of $\boldsymbol{c}$, and λ_v denotes parameters of a GV PDF. The second term of Eq. (2) can be viewed as a penalty term for over-smoothing. The use of this algorithm dramatically reduces the buzziness in synthesized speech and improves the speech quality [31]. This was one of main components of Nitech's Blizzard Challenge 2005 system [34].

3.3.3 CSMAPLR

The MLLR adaptation algorithms utilize the ML criterion to estimate linear transformation matrices. However, the amount of adaptation data is usually very limited in the adaptation stage. Therefore, we should use more robust criteria such as the MAP criterion. In the MAP estimation, we estimate the linear transformation matrices $\boldsymbol{X}$ as follows:

$$\hat{\boldsymbol{X}} = \arg\max_{\boldsymbol{X}} P(\boldsymbol{o} \mid \lambda, \boldsymbol{X}) P(\boldsymbol{X}) \tag{3}$$

where $P(\boldsymbol{X})$ is a prior distribution for the linear transformation matrix $\boldsymbol{X}$.

In the structured MAP linear regression (SMAPLR) [41], tree structures of the distributions effectively cope with the control of the parameters of prior distributions. Specifically, we first estimate a global linear transformation matrix at the root node of the tree structure using all the adaptation data, and then propagate it to its child nodes as their prior distributions. In the child nodes, linear transformation matrices are estimated again using their adaptation data, based on the MAP criterion with the propagated prior distributions. Then, the recursive MAP-based estimation of the transformation matrices from a root node to lower nodes is conducted. Nakano et al. applied the SMAP to the constrained MLLR adaptation and derived constrained SMAPLR (CSMAPLR) [32], in which the linear transformation matrices for both mean vectors and covariance matrices of state output PDFs are shared and estimated using the recursive MAP criterion. The CSMAPLR adaptation algorithm can utilize the tree structure more effectively than the constrained MLLR adaptation since the tree structure represents connection and similarity between the distributions, and the propagated prior information automatically reflects the connection and similarity. This algorithm was applied to the HMM-based speech synthesis and showed that it was better than the other linear transformation-based adaptation techniques [32]. We expect that it is also useful for speech recognition.

3.3.4 hts_engine API

Since version 1.1, a small stand-alone run-time synthesis engine named `hts_engine` has been included in the HTS releases. It works without the HTK libraries, hence it is free from the HTK license. Users can develop their own open or proprietary software based on the run-time synthesis engine. In fact, a part of `hts_engine` has been integrated into several softwares, such as ATR XIMERA [42], Festival [18], and OpenMARY [43]. The spectrum and prosody prediction modules of ATR XIMERA are based on `hts_engine`. Festival includes `hts_engine` as its one of the waveform synthesis modules. The upcoming version of OpenMARY uses the JAVA version of `hts_engine`.

As described above, `hts_engine` has been used as a module rather than a stand-alone software. It suggests that users demand `hts_engine` library, not stand-alone program. In response to this demand, we decided to rewrite `hts_engine` to API-style implementation. We released the β version of this API at the same time we released HTS version 2.0.1. It is written in C and provides various functions required to setup and drive the synthesis engine. Reference of this API is available at the HTS website [5]. The stable version, `hts_engine API` version 1.0, will be released with HTS version 2.1. It will support LSP-type parameters in addition to cepstral parameters. The speech parameter generation algorithm considering GV will also be included.

3.4 Demonstrations and documentation

Currently two demo scripts to construct speaker-dependent systems (English and Japanese) and a demo script to train a speaker-adaptation system (English) are released. The English demo scripts use the CMU ARCTIC databases and generate model files for Festival and `hts_engine`. The Japanese demo script uses the Nitech database and generates model files for the Galatea toolkit [20]. These scripts demonstrate the training processes and the functions of HTS. Six voices for Festival trained by the CMU ARCTIC databases have also been released. Each HTS voice consists of model files trained by the demo script, and can be used as a voice for Festival without any other HTS tools.

Currently no documentation for HTS is available. However, the interface and functions of HTS are almost the same as those of HTK. Therefore, users who are familiar with HTK can easily understand how to use HTS. The manual of HTK [15] is also very useful. There is also an open mailing list for the discussion of HTS (`hts-users@sp.nitech.ac.jp`).

4. Other applications

Although HTS has been developed to provide a research platform for HMM-based speech synthesis, it has also been used in various other ways, such as

- Human motion synthesis [44–46],
- Face animation synthesis [47],
- Audio-visual synthesis [48, 49] and recognition [50],
- Acoustic-articulatory inversion mapping [51],
- Prosodic event recognition [52, 53],
- Mispronunciation detection in CALL systems [54],
- Very low-bitrate speech coder [55],
- Acoustic model adaptation for coded speech [56],
- Training data generation for ASR systems [57].
- Automatic evaluation of ASR systems [58].
- Online handwriting recognition [59].

We hope that HTS will contribute to progress in other research fields as well as speech synthesis.

5. Conclusions and future release plans

This report described the recent developments of the HMM-based speech synthesis system (HTS). Internally, we have a number of variants of HTS, e.g.,

- Variational Bayes [60],
- Trajectory HMMs [61],
- Minimum generation error training [62],
- Shared tree construction [63],
- Eigenvoice [4],
- Multiple linear regression HMMs [64].

Hopefully, we can integrate valuable features of these variants into future HTS releases. The current release plan is as follows:

- **Version 2.1α (November 2007)**
  Speech parameter generation algorithm considering GV.
- **Version 2.1β (January 2008)**
  HSMM training and generation.
- **Version 2.1 (March 2008)**
  Advanced adaptation techniques.

On-line demonstrations which have been built using the above HTS version 2.1 features are available at [65].

6. Acknowledgment

The authors are grateful to all people who contributed to the development of HTS. This work was partly supported by the MEXT e-Society project.

文 献

[1] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, “Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis,” Proc. Eurospeech, pp.2347–2350, 1999.

[2] J. Yamagishi, Average-Voice-Based Speech Synthesis, Ph.D. thesis, Tokyo Institute of Technology, 2006.

[3] T. Yoshimura, T. Masuko, K. Tokuda, T. Kobayashi, and T. Kitamura, “Speaker interpolation for HMM-based speech synthesis system,” J. Acoust. Soc. Jpn. (E), vol.21, no.4, pp.199–206, 2000.

[4] K. Shichiri, A. Sawabe, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, “Eigenvoices for HMM-based speech synthesis,” Proc. ICSLP, pp.1269–1272, 2002.

[5] K. Tokuda, H. Zen, J. Yamagishi, T. Masuko, S. Sako, A. Black, and T. Nose, “The HMM-based speech synthesis system (HTS).” `http://hts.sp.nitech.ac.jp/`.

[6] A. Hunt and A. Black, “Unit selection in a concatenative speech synthesis system using a large speech database,” Proc. ICASSP, pp.373–376, 1996.

[7] N. Mizutani, K. Tokuda, and T. Kitamura, “Concatenative speech synthesis based on HMM,” Proc. Autumn Meeting of ASJ, pp.241–242, 2002. (in Japanese).

[8] C. Allauzen, M. Mohri, and M. Riley, “Statistical modeling for unit selection in speech synthesis,” Proc. the 42nd meeting of the ACL, 2004.

[9] S. Sakai and H. Shu, “A probabilistic approach to unit selection for corpus-based speech synthesis,” Proc. Interspeech (Eurospeech), pp.81–84, 2005.

[10] Z.H. Ling and R.H. Wang, “HMM-based unit selection using frame sized speech segments,” Proc. Interspeech (ICSLP), pp.2034–2037, 2006.

[11] C. Weiss and W. Hess, “Conditional random fields for hierarchical segment selection in text-to-speech synthesis,” Proc. Interspeech (ICSLP), pp.1090–1093, 2006.

[12] K. Tokuda, T. Masuko, N. Miyazaki, and T. Kobayashi, “Multi-space probability distribution HMM,” IEICE Trans. Inf. & Syst., vol.E85-D, no.3, pp.455–464, Mar. 2002.

[13] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, “Speech parameter generation algorithms for HMM-based speech synthesis,” Proc. ICASSP, pp.1315–1318, 2000.

[14] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, “Speaker interpolation in HMM-based speech synthesis system,” Proc. Eurospeech, pp.2523–2526, 1997.

[15] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X.Y. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, The Hidden Markov Model Toolkit (HTK) version 3.4, 2006. `http://htk.eng.cam.ac.uk/`.

[16] K. Shinoda and T. Watanabe, “MDL-based context-dependent subword modeling for speech recognition,” J. Acoust. Soc. Jpn.(E), vol.21, no.2, pp.79–86, 2000.

[17] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, “Duration modeling for HMM-based speech synthesis,” Proc. ICSLP, pp.29–32, 1998.

[18] A. Black, P. Taylor, and R. Caley, “The festival speech synthesis system.” `http://www.festvox.org/festival/`.

[19] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, “Incorporation of mixed excitation model and postfilter into HMM-based text-to-speech synthesis,” IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J87-D-II, no.8, pp.1563–1571, Aug. 2004.

[20] “Galatea – An open-source toolkit for anthropomorphic spoken dialogue agent.” `http://hil.t.u-tokyo.ac.jp/galatea/`.

[21] D. Huggins-Daines and A. Rudnicky, “A constrained Baum-Welch algorithm for improved phoneme segmentation and efficient training,” Proc. of Interspeech, pp.1205–1208, 2006.

[22] K. Tokuda, H. Zen, and T. Kitamura, “Reformulating the HMM as a trajectory model,” Proc. Beyond HMM – Workshop on statistical modeling approach for speech recognition, 2004.

[23] M. Gales, “Maximum likelihood linear transformations for HMM-

based speech recognition," Computer Speech & Language, vol.12, no.2, pp.75–98, 1998.

[24] M. Gales, "Semi-tied covariance matrices for hidden Markov models," IEEE Transactions on Speech and Audio Processing, vol.7, no.3, pp.272–281, 1999.

[25] M. Gales, "Maximum likelihood multiple projection schemes for hidden Markov models," IEEE Trans. Speech & Audio Process., vol.10, no.2, pp.37–47, 2002.

[26] J. Yamagishi, M. Tachibana, T. Masuko, and T. Kobayashi, "Speaking style adaptation using context clustering decision tree for HMM-based speech synthesis," Proc. ICASSP, pp.5–8, 2004.

[27] J. Gauvain and C.H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," IEEE Trans. on Speech & Audio Process., vol.2, no.2, pp.291–298, 1994.

[28] L. Qin, Y.J. Wu, Z.H. Ling, and R.H. Wang, "Improving the performance of HMM-based voice conversion using context clustering decision tree and appropriate regression matrix," Proc. of Interspeech (ICSLP), pp.2250–2253, 2006.

[29] H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "A hidden semi-Markov model-based speech synthesis system," IEICE Trans. Inf. & Syst., vol.E90-D, no.5, pp.825–834, 2007.

[30] J. Yamagishi and T. Kobayashi, "Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training," IEICE Trans. Inf. & Syst., vol.E90-D, no.2, pp.533–543, 2007.

[31] T. Toda and K. Tokuda, "A speech parameter generation algorithm considering global variance for HMM-based speech synthesis," IEICE Trans. Inf. & Syst., vol.E90-D, no.5, pp.816–824, 2007.

[32] Y. Nakano, M. Tachibana, J. Yamagishi, and T. Kobayashi, "Constrained structural maximum a posteriori linear regression for average-voice-based speech synthesis," Proc. Interspeech, pp.2286–2289, 2006.

[33] H. Kawahara, I. Masuda-Katsuse, and A. Cheveigné, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F_0 extraction: possible role of a repetitive structure in sounds," Speech Communication, vol.27, pp.187–207, 1999.

[34] H. Zen, T. Toda, M. Nakamura, and K. Tokuda, "Details of Nitech HMM-based speech synthesis system for the Blizzard Challenge 2005," IEICE Trans. Inf. & Syst., vol.E90-D, no.1, pp.325–333, Jan. 2007.

[35] H. Zen, T. Toda, and K. Tokuda, "The Nitech-NAIST HMM-based speech synthesis system for the Blizzard Challenge 2006," Blizzard Challenge Workshop, 2006.

[36] J. Yamagishi, H. Zen, T. Toda, and K. Tokuda, "Speaker-independent HMM-based speech synthesis system – HTS-2007 system for the Blizzard Challenge 2007," Proc. Blizzard Challenge 2007, 2007.

[37] M. Russell and R. Moore, "Explicit modeling of state occupancy in hidden Markov models for automatic speech recognition," Proc. ICASSP, pp.5–8, 1985.

[38] M. Tachibana, J. Yamagishi, T. Masuko, and T. Kobayashi, "A style adaptation technique for speech synthesis using HSMM and suprasegmental features," IEICE Trans. Inf. & Syst., vol.E89-D, no.3, pp.1092–1099, 2006.

[39] K. Oura, H. Zen, Y. Nankaku, A. Lee, and K. Tokuda, "Hidden semi-Markov model based speech recognition system using weighted finite-state transducer," Proc. ICASSP, pp.33–36, 2006.

[40] K. Shinoda and C.H. Lee, "A structural Bayes approach to speaker adaptation," IEEE Trans. Speech & Audio Process., vol.9, pp.276–287, 2001.

[41] O. Shiohan, Y. Myrvoll, and C.H. Lee, "Structural maximum a posteriori linear regression for fast HMM adaptation," Computer Speech & Language, vol.16, no.3, pp.5–24, 2002.

[42] H. Kawai, T. Toda, J. Yamagishi, T. Hirai, J. Ni, T. Nishizawa, M. Tsuzaki, and K. Tokuda, "XIMERA: A concatenative speech synthesis system with large scale corpora," IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J89-D, no.12, pp.2688–2698, Dec. 2006.

[43] M. Schröder and J. Trouvain, "The German text-to-speech synthesis system MARY: A tool for research, development and teaching," International Journal of Speech Technology, vol.6, pp.365–377, 2003.

[44] K. Mori, Y. Nankaku, C. Miyajima, K. Tokuda, and T. Kitamura, "Motion generation for Japanese finger language based on hidden Markov models," Proc. FIT, pp.569–570, 2005. (in Japanese).

[45] N. Niwase, J. Yamagishi, and T. Kobayashi, "Human walking motion synthesis with desired pace and stride length based on HSMM," IEICE Trans. Inf. & Syst., vol.E88-D, no.11, pp.2492–2499, 2005.

[46] G. Hofer, H. Shimodaira, and J. Yamagishi, "Speech driven head motion synthesis based on a trajectory model," Proc. SIGGRAPH, 2007.

[47] O. Govokhina, G. Bailly, G. Breton, and P. Bagshaw, "TDA: a new trainable trajectory formation system for facial animation," Proc. Interspeech, pp.1274–1247, 2006.

[48] M. Tamura, S. Kondo, T. Masuko, and T. Kobayashi, "Text-to-audio-visual speech synthesis based on parameter generation from HMM," Proc. Eurospeech, pp.959–962, 1999.

[49] S. Sako, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "HMM-based text-to-audio-visual speech synthesis," Proc. ICSLP, pp.25–28, 2000.

[50] T. Ishikawa, Y. Sawada, H. Zen, Y. Nankaku, C. Miyajima, K. Tokuda, and T. Kitamura, "Audio-visual large vocabulary continuous speech recognition based on early integration," Proc. FIT, pp.203–204, 2002. (in Japanese).

[51] K. Richmond, "A trajectory mixture density network for the acoustic-articulatory inversion mapping," Proc. of Interspeech, pp.577–580, 2006.

[52] K. Emoto, H. Zen, K. Tokuda, and T. Kitamura, "Accent type recognition for automatic prosodic labeling," Proc. Autumn Meeting of ASJ, pp.225–226, 2003. (in Japanese).

[53] H.L. Wang, Y. Qian, F. Soong, J.L. Zhou, and J.Q. Han, "A multi-space distribution (MSD) approach to speech recognition of tonal languages," Proc. of Interspeech, pp.125–128, 2006.

[54] L. Zhang, C. Huang, M. Chu, F. Soong, X. Zhang, and Y. Chen, "Automatic detection of tone mispronunciation in Mandarin," Proc. ISCSLP, pp.590–601, 2006.

[55] T. Hoshiya, S. Sako, H. Zen, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Improving the performance of HMM-based very low bitrate speech coding," Proc. ICASSP, pp.800–803, 2003.

[56] K. Tanaka, S. Kuroiwa, S. Tsuge, and F. Ren, "An acoustic model adaptation using HMM-based speech synthesis," Proc. NLPKE, pp.368–373, 2003.

[57] M. Ishihara, C. Miyajima, N. Kitaoka, K. Itou, and K. Takeda, "An approach for training acoustic models based on the vocabulary of the target speech recognition task," Proc. Spring Meeting of ASJ, pp.153–154, 2007. (in Japanese).

[58] R. Terashima, T. Yoshimura, T. Wakita, K. Tokuda, and T. Kitamura, "An evaluation method of ASR performance by HMM-based speech synthesis," Proc. Spring Meeting of ASJ, pp.159–160, 2003. (in Japanese).

[59] L. Ma, Y.J. Wu, P. Liu, and F. Soong, "A MSD-HMM approach to pen trajectory modeling for online handwriting recognition," Proc. ICDAR, pp.128–132, 2007.

[60] Y. Nankaku, H. Zen, K. Tokuda, T. Kitamura, and T. Masuko, "A Bayesian approach to HMM-based speech synthesis," Tech. rep. of IEICE, pp.19–24, 2003. (in Japanese).

[61] H. Zen, K. Tokuda, and T. Kitamura, "Reformulating the HMM as a trajectory model by imposing explicit relationships between static and dynamic feature vector sequences," Computer Speech & Language, vol.21, no.1, pp.153–173, 2006.

[62] Y.J. Wu and R.H. Wang, "Minimum generation error training for HMM-based speech synthesis," Proc. ICASSP, pp.89–92, 2006.

[63] J. Yamagishi, M. Tamura, T. Masuko, K. Tokuda, and T. Kobayashi, "A context clustering technique for average voice models," IEICE Trans. Inf. & Syst., vol.E86-D, no.3, pp.534–542, 2003.

[64] T. Nose, J. Yamagishi, T. Masuko, and T. Kobayashi, "A style control technique for HMM-based expressive speech synthesis," IEICE Trans. Inf. & Syst., vol.E90-D, no.9, pp.1406–1413, 2007.

[65] http://www.cstr.ed.ac.uk/projects/festival/onlinedemo.html.