Temporally variable multi-aspect auditory morphing enabling extrapolation without objective and perceptual breakdown

Hideki Kawahara, Masanori Morise
Ryuichi Nisimura, Toshio Irino and Hideki Banno
Wakayama University, Meijo University, Japan
Outline

- What is the problem?
  - Demos for introducing backgrounds (TANDEM-STRAIGHT and morphing) and the problem
- New formulation of morphing
  - Morphing on the fly
  - Morphing for post production
- Implementation
- Demo
- Conclusion
Outline

- What is the problem?
  - Demos for introducing backgrounds (TANDEM–STRAIGHT and morphing) and the problem

- New formulation of morphing
  - Morphing on the fly
  - Morphing for post production

- Implementation

- Demo

- Conclusion
What is the problem? (demos)

- **TANDEM–STRAIGHT**
  - Background (ICASSP2008)
    - Complete reformulation of STRAIGHT using TANDEM power spectrum estimation method for periodic signals and consistent sampling framework

- **Singing voice morphing (v.morish)**
  - How to implement temporally variable morphing of time axis?

- **CrestMuse project**
  - Please search Youtube
TANDEM–STRAIGHT (movie)
Complete re-formulation of STRAIGHT

No code is re-used

\[ P_T(\omega, t) = |S(\omega, t - T_0/4)|^2 + |S(\omega, t + T_0/4)|^2 \]

\[ C(\omega) = \int_{\omega_L}^{\omega} P_T(\lambda) d\lambda \]

\[ L_S(\omega) = \ln \left[ C(\omega + \omega_0/2) - C(\omega - \omega_0/2) \right] - \ln \omega_0 \]

\[ P_{TST}(\omega) = e^{[\bar{q}_1 (L_S(\omega-\omega_0)+L_S(\omega+\omega_0)) + \bar{q}_0 L_S(\omega)]} \]
Complete re-formulation of STRAIGHT

No code is re-used

\[ P_{T}(\omega, t) = |S(\omega, t - T_0/4)|^2 + |S(\omega, t + T_0/4)|^2 \]

\[ C(\omega) = \int_{\omega_L}^{\omega} P_{T}(\lambda) d\lambda \]

\[ L_S(\omega) = \ln[C(\omega + \omega_0/2) - C(\omega - \omega_0/2)] - \ln \omega_0 \]

\[ P_{TST}(\omega) = e^{[\bar{q}_1 (L_S(\omega-\omega_0) + L_S(\omega+\omega_0)) + \bar{q}_0 L_S(\omega)]} \]

D/A conversion based on consistent sampling (with approximation)
Complete re-formulation of STRAIGHT
No code is re-used
TANDEM–STRAIGHT

- Complete re-formulation of STRAIGHT
- No code is re-used

```c
quefrenyAxis = [0:fftl/2, -(fftl/2-1:-1:1)]/fs;
tmpLogSpectrum = log(smoothPsgram( : ));
tmpCepstrum = real(fft([tmpLogSpectrum;tmpLogSpectrum(end-1:-1:2)]));
normalizedQuefreny = min(2,2*abs(quefrenyAxis*f0));
lifter = 1.0-q1*(1-cos(pi*normalizedQuefreny))';
tmpFixedPsgram = exp(real(ifft(tmpCepstrum.*lifter)));
```
Temporally variable morphing

- V.morish (demo)
What is the problem? (continued)

- Incremental
  - Real-time applications
  - Monotonic
  - Extrapolation

No
Not always

\[ T_{Am}(x) = (1 - r)T_{AA} + rT_{AB} \]
What is the problem? (continued)
What is the problem? (continued)

interpolation
What is the problem? (continued)

- Break down
- Interpolation
- Extrapolation
- Non-monotonic mapping
What is the problem? (continued)

- Incremental
  - Real-time applications: Yes
- Monotonic
  - Extrapolation: Not always

\[
T_{Am}(x) = \int_0^x \left[ (1 - r(\lambda)) \left( \frac{dT_{AA}(\lambda)}{d\lambda} \right) + r(\lambda) \left( \frac{dT_{AB}(\lambda)}{d\lambda} \right) \right] d\lambda
\]
Outline

- What is the problem?
  - Demos for introducing backgrounds (TANDEM–STRAIGHT and morphing) and the problem

- New formulation of morphing
  - Morphing on the fly
  - Morphing for post production

- Implementation

- Demo

- Conclusion
New formulation of morphing

- Incremental
  - Real-time applications
  - Yes

- Monotonic
  - Extrapolation
  - Yes

\[ T_{Am}(x) = \int_0^x \exp \left[ (1 - r(\lambda)) \log \left( \frac{dT_{AA}(\lambda)}{d\lambda} \right) + r(\lambda) \log \left( \frac{dT_{AB}(\lambda)}{d\lambda} \right) \right] d\lambda \]
New formulation of morphing

- Incremental
  - Real-time applications: **Yes**
- Monotonic
  - Extrapolation: **Yes**

\[
T_{Am}(x) = \int_0^x \exp \left[ (1 - r(\lambda)) \log \left( \frac{dT_{AA}(\lambda)}{d\lambda} \right) + r(\lambda) \log \left( \frac{dT_{AB}(\lambda)}{d\lambda} \right) \right] d\lambda
\]

\[= 0\]
New formulation of morphing

- Incremental
  - Real-time applications: Yes

- Monotonic
  - Extrapolation: Yes

\[ T_{Am}(x) = \int_0^x \left( \frac{dT_{AB}(\lambda)}{d\lambda} \right)^r(\lambda) \, d\lambda \]
Morphing on the fly

- Morphing rate is given in real-time

Time in the real world

\[ t_s = \int_0^{t_s} d\lambda, \]

Time in utterance A

\[ T_{SA}(t_s) = \int_0^{t_s} \left( \frac{dT_{AB}(T_{SA}(\lambda))}{d\lambda} \right)^{-r_{AB}^{(t)}(\lambda)} d\lambda, \]

Time in utterance B

\[ T_{SB}(t_s) = \int_0^{t_s} \left( \frac{dT_{BA}(T_{SB}(\lambda))}{d\lambda} \right)^{r_{AB}^{(t)}(\lambda) - 1} d\lambda, \]
Morphing on for post production

- Morphing rate is given on a reference time

\[
\int_0^{t_s} \omega_s(\lambda) d\lambda
\]

excitation pulse

“real” time

“reference” time

time-A
time-B

\[T_{rs} \quad T_{st} \quad r_{AB} \quad \Theta_B \quad T_{rA} \quad \Theta_A \quad T_{rB}\]
Outline

- What is the problem?
  - Demos for introducing backgrounds (TANDEM–STRAIGHT and morphing) and the problem
- New formulation of morphing
  - Morphing on the fly
  - Morphing for post production
- Implementation
- Demo
- Conclusion
Outline

- What is the problem?
  - Demos for introducing backgrounds (TANDEM–STRAIGHT and morphing) and the problem
- New formulation of morphing
  - Morphing on the fly
  - Morphing for post production
- Implementation
- Demo
- Conclusion
Demo (morphing using a new GUI)

- Morphing for post production
Conclusions

- Morphing re-formulation based on log-of-derivative of mapping eliminates breakdown in extrapolation
- Temporally variable morphing is formulated for real-time manipulation as well as for post production applications
- GUI as well as API are available
- Completely independent from (so called) legacy-STRAIGHT
- Will be available in a couple of weeks