2011 IEEE/RSJ International Conference on
Intelligent Robots and Systems
September 25-30, 2011. San Francisco, CA, USA

VocaWatcher: Natural Singing Motion Generator for a Humanoid Robot

Shuuji Kajita, Tomoyasu Nakano, Masataka Goto, Yosuke Matsusaka, Shin'ichiro Nakaoka, and Kazuhito Yokoi

Abstract— **In this paper, we describe VocaWatcher, a novel robot motion generator that enables a humanoid robot to sing with realistic facial expressions and naturally synthesized singing voices. This robot singer is an important and attractive humanoid robot application for the entertainment scene; moreover, it promotes state-of-the-art integration of robot engineering, music processing, and image processing. To overcome the difficulties of generating natural facial expressions that are precisely synchronized with singing voices, VocaWatcher imitates a human singer by analyzing a video clip of a human singing, recorded by a single video camera. VocaWatcher can control mouth, eye, and neck motions by imitating the corresponding human movements, which are estimated without using any markers in the video. It can also synthesize singing voices by imitating the pitch and dynamics of the human singing in the same video.**

I. INTRODUCTION

Since Kato's WABOT-2 [1] played an electronic organ in 1985, music has been an important and attractive application for humanoid robots. Humanoid robots have been developed that play various instruments, including a flute [2] and a theremin [3]. Singing humanoid robots have also been developed with synthesized singing voices; however, such robots [4], [5] do not appear to be natural because of the limitations of manual control. A humanoid robot produced by Murata, Nakadai, *et al.*[5] sang and moved in time to musical beats, using a real-time beat tracking technique. However, none of these robots could generate realistic facial expressions synchronized with a naturally synthesized singing voice.

We previously demonstrated a singing humanoid robot in 2009; however, its facial expressions and singing voice were neither realistic nor natural. This robot, *Cybernetic Human HRP-4C* (Fig. 1), was designed so that its dimensions (158 cm height and 43 kg weight with 42 DOF) closely resembled those of an average young Japanese female [6]. HRP-4C had the advantage of eight motors in its small head allowing realistic facial expressions; however, it was not easy to control those motors in a natural way. Our first demonstration of HRP-4C singing was at the public exhibition CEATEC JAPAN 2009 [7], which was achieved in collaboration with the Yamaha Corporation. Facial expressions and neck motions were generated by well-known techniques, such as key poses and semi-automatic motion generation. Singing voices were synthesized using Vocaloid2, a commercial software system developed by the Yamaha Corporation [12]. Our demonstration was successful; however, there was much room for improvement, which motivated the work presented in this paper.

Shuuji Kajita, Tomoyasu Nakano, Masataka Goto, Yosuke Matsusaka, Shin'ichiro Nakaoka, and Kazuhito Yokoi are with the National Institute of Advanced Industrial Science and Technology (AIST), 1-1-1 Umezono, Tsukuba, Ibaraki 305-8568, Japan (e-mail: s.kajita@aist.go.jp).

Fig. 1. A humanoid robot named *Cybernetic Human HRP-4C*.

A promising solution to the problem of generating realistic humanoid robot motions is the imitation of human motions, as shown in computer graphics. Wilbers, Ishi, and Ishiguro [8] used a motion capture technique to imitate a human face wearing 31 reflective markers, which controlled the face of their android Repliee Q2 that wore the same number of markers. Jaeckel, Campbell, and Melhuish [9] also controlled the face of their humanoid robot using video input from a human face; however, they did not use markers. Instead, they used person-specific Active Appearance Models (AAMs) fitted to video frames when tracking facial movements. However, the use of AAMs required training data. Furthermore, audio signal processing did not help facial control in previous studies.

We propose a robot motion generation method, which we refer to as *VocaWatcher*, using both image and music signal processing to produce a realistic singing performance with the HRP-4C robot. When provided with a recorded video clip of a singing performance by a human singer, our robot can sing naturally by synthesizing the singing voice imitating the pitch and dynamics of the human singing in the clip. The robot also generates synchronous mouth, eye, and neck motions to imitate the facial expressions of the

978-1-61284-455-8/11/$26.00 ©2011 IEEE

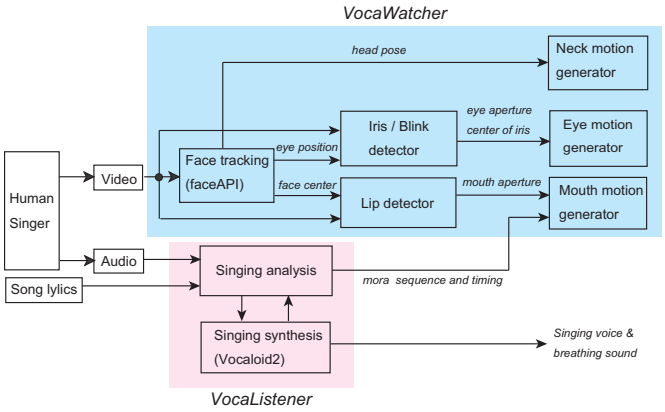


Fig. 2. An overview of our singing robot system which consists of two subsystems: *VocaWatcher* and *VocaListener*.

same human who is singing. VocaWatcher requires neither facial markers nor multiple cameras and it can utilize audio-based timing information that is estimated by analyzing the human's singing voice to help generate the robot's motions.

II. Overview of the Singing Robot System

Our singing robot system uses the HRP-4C and consists of two subsystems: *VocaWatcher* for generating singing motions and *VocaListener* for synthesizing singing voices, as shown in Fig.2. These subsystems analyze audio signals and video frames in a video clip of a human singer, and imitate the human singing. Note that analysis, generation, and synthesis are not executed in real time; nonetheless, the robot can play back the synthesized singing voice along with the synchronized motions.

VocaListener [11] analyzes the audio signals of the singing voice in the video clip with the help of the written text of its lyrics. The system automatically identifies each musical note produced by the singing voice and estimates expressive information, consisting of the pitch and dynamics, to synthesize a robot singing voice with a variety of voice timbres. The main advantages of VocaListener are that its singing voice appears to be highly natural without laborious manual adjustments and that the synthesized voice timbre can be changed easily. In this paper, we extended VocaListener to imitate breathing sounds that make the robot singing more realistic.

VocaWatcher analyzes the human face and head in the video clip frames, but without any markers. VocaWatcher automatically detects the head position and rotation, the iris and eyelid, and the mouth aperture, to generate neck, eye, and mouth motions, using the vowel sequence and precise timing information provided by VocaListener. The main advantages of VocaWatcher are highly realistic motion generation without laborious manual adjustment and precise synchronization of motions with the synthesized singing voice.

Video clips of recorded singing performances were used to demonstrate our system (Fig.3). In the video clips, each of three different female singers sang three different songs

Fig. 3. A recording scene with the target human singing: a female singer on the right was recorded using a home video camera on the top left.

having different moods. For example, one of the songs is entitled "PROLOGUE" (song number: RWC-MDB-P-2001 No.7) and it was taken from the RWC Music Database (Popular Music) [10], which is a copyright-cleared music database utilized by researchers worldwide (more than 300 institutes) as a common resource for music research. During recordings of singing performances, a singer stood in front of the microphone and a fixed camera (Fig.3). The singer was allowed to move her head and hands naturally, but she was asked to stay within a specified area to allow the upper half of her body to be recorded.

III. Singing Synthesis from Human Voice

A. VocaListener: An automatic parameter estimation system for singing synthesis by using a singing voice and its lyrics

VocaListener [11] iteratively estimates the parameters for pitch (F_0) and the dynamics (power) for a singing synthesis system (*e.g.,* Yamaha's Vocaloid [12]) to synthesize singing similar to a human singer (**Fig. 4**). The singing voice of a human is natural, and the synthesized singing voice imitates it to produce a human-like and natural output without any time-consuming manual adjustments. Iterative estimation provides robustness allowing the use of different singing synthesis systems and different singer databases. The mean error after iterations was much smaller than the previous method [13] (see [11] for details)[1].

VocaListener also provides a highly accurate lyrics-to-singing synchronization function. Given a singing voice and the corresponding lyrics without any score information, VocaListener automatically synchronizes them and determines where each musical note corresponds to a mora[2]in the lyrics. We also provide an interface for easily correcting errors simply by pointing them out.

B. Extending VocaListener to imitate breathing

A singer has to breathe while singing; thus, the mouth of the robot made facial motions to imitate changes in a singer's

[1]Demonstration videos including examples of synthesized singing are available at http://staff.aist.go.jp/t.nakano/VocaListener/

[2]Mora is a partly decomposed unit of syllables. Each mora of Japanese pronunciation is mapped into a musical note, where the mora representation can be classified into three types: "V", "CV", and "N". "V" denotes vowel, "C" denotes consonant, and "N" denotes syllabic nasal.

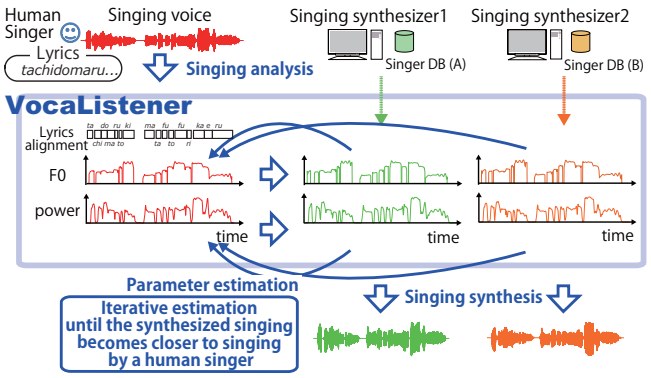


Fig. 4. Overview of VocaListener, which iteratively estimates parameters of pitch and dynamics for singing synthesis from the human singing voice and the song lyrics.

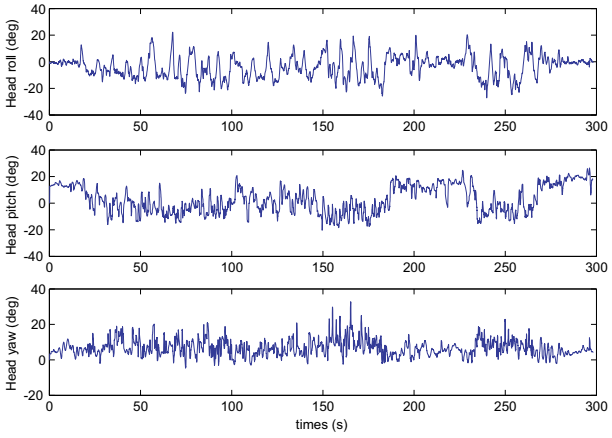


Fig. 5. Singer head posture detected by faceAPI.

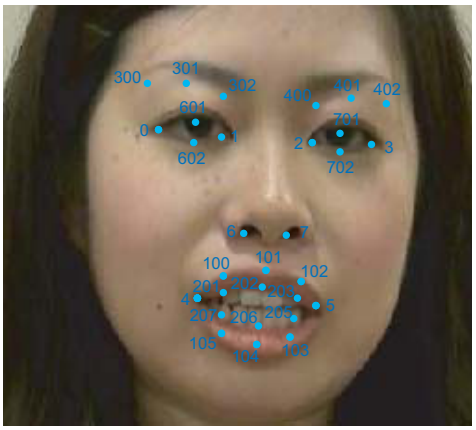


Fig. 6. Feature points (face landmarks) detected by faceAPI. They are used as reference information for our additional image processing.

facial expressions during breathing (see IV-D). It would have appeared unnatural if there was no breathing sound during these mouth-opening motions. We addressed this issue by extending VocaListener to synthesize breathing sounds by imitating the breathing sounds of a human singer, because the original VocaListener did not imitate breathing sounds.

Our automatic breathing detection method [14] estimated the onset time and duration of every breathing sound produced by a human singer. This method detected almost all the breathing sounds, although non-breathing sounds were sometimes wrongly detected as breathing sounds. We improve breathing detection performance by developing a new technique to recover from errors by eliminating detected sounds that were not located immediately before a phrase, and whose length was less than 50 ms or more than 1225 ms. This range was identified by our previous study investigating the length of breathing sounds [14]. Here a phrase was defined as a non-silent section in the singing voice synthesized by VocaListener. If an error persisted after the improvement technique, it could be manually corrected.

Each breathing sound detected in human singing was imitated using our new breathing synthesis system, which is based on the speech manipulation system *TANDEM-STRAIGHT* [15]. We required the breathing sound of the target singer; hence, we used the same singing synthesis system (i.e., Vocaloid) to synthesize an example of a breathing sound. The synthesized example was analyzed by TANDEM-STRAIGHT to obtain its spectral envelope. The duration and power of the spectral envelope were then controlled to ensure that the duration and power of the synthesized breathing sound were similar to each breathing sound detected in the human singing.

IV. Face Motion Analysis

Face motion analysis used commercial face tracking software and our original image processing method for the eyelid aperture, iris position, and mouth aperture. These results were not directly mapped to the robot motion; instead, they were used with a modification as explained in Section V.

A. Face tracking

We used video data with 960×540 pixels and 29.97 frames per second (FPS). The first step was to process the video, for which we used commercial face tracking software, *faceAPI*, produced by Seeing Machines Inc.[16]. This system tracks a human face in the video and estimates the head position and rotation in 3D space. Fig.5 shows the head rotation of the performer from the start to the end of a song of 298.2 s length. The same software also detected feature points on the face (face landmarks) for every frame of the video (Fig.6). For example, the points numbered $0, 601, 1, 602$ correspond to the right eye's area. Unfortunately, these points did not reflect the eyelid aperture and the distance between points 601 and 602 never changed, even when the right eyelid was closed.

This meant we had to perform further image processing to detect the eyelid aperture, which is explained in the next section.

B. Background of iris and eyelid detection

For many years, iris and eyelid detection has been developed for use in different applications. For example, Mat-

sumoto *et al.*[17] developed an algorithm for detecting gaze direction and produced a gaze-controlled wheel chair. Morris *et al.*[18] developed an algorithm to detect eye blinking and produced a command interface for people with paralysis. However, no previous research has investigated iris and eyelid detection for a singing face. Iris and eyelid detection for singing face presents the following new problems.

A singer might maintain a narrowing of his[her] eyes for long periods during a performance. Thus, we had to continuously detect the aperture ratio of the eyelid in each frame, whereas conventional methods only detect eye blinks as discrete values.

A singer might continuously move his[her] head during a performance. Thus, we had to capture the whole image of the face from a distant position, because we could not force a singer to wear an eye camera simply to obtain a high-resolution image of the eye.

To deal these problems, we developed a novel iris and eyelid detection algorithm for use during singing. Our algorithm has the following advantages:

- Detect the iris, even if the eyelid is half closed.
- Detect the aperture ratio of the eyelid as a continuous value.
- Detect the above information, even from a low-resolution image.

We explain the algorithm in the next.

C. Algorithm for iris and eyelid detection

During iris and eyelid detection, we first focus on the eye region of the input image, which is detected using faceAPI (the rectangular area supported by feature points 0, 601, 1, 602 as shown in Fig.6).

Fig.7 shows an overview of the iris and eyelid detection process.

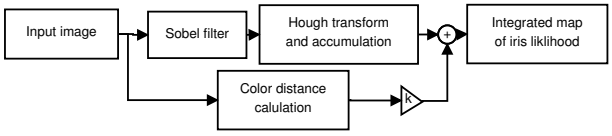


Fig. 7. Overview of the iris and eyelid detection process.

Our algorithm consists of following parts:

Enhanced Hough voting with color distance: The iris of the singer is occluded by a half-closed eyelid for a long period during the performance. In order to ensure robust iris detection, we developed an algorithm that uses a regular Hough transform-based voting map, and integrates color similarity of the iris in voting. The algorithm is formalized as follows:

$$L = A + kD \quad (1)$$

$$A_{x+a,y+b} \leftarrow A_{x+a,y+b} + |\nabla I| \quad (2)$$
$$(a = r\sin\theta, b = r\cos\theta)$$

$$|\nabla I| = (dI/dx + dI/dy)^{1/2} \quad (3)$$

$$D_{xy} = (1 - I_{xy} - I^r)^2, \quad (4)$$

where I is the input image, $|\nabla I|$ is the edge image detected by a Sobel operator, A is the accumulated voting result of the circular Hough transform, D is the color distance, I^r is the iris color, k is an integration constant, and L is a 2D map of the iris likelihood. In this particular experiment, we used color distance calculated from a gray scale image where the iris was assumed to be black. We consider that the color distance measure can be easily extended to a multi-scale for detecting non-black irises.

Finally, the position of the iris, p, is estimated using the iris likelihood L, as follows.

$$p = \arg\max_{xy} L_{xy} \quad (5)$$

Eyelid aperture rate detection uses the subpixel algorithm: In most of source images, the height of the eye is about 3-6 pixels. If we use a conventional pixel-based algorithm, the eyelid aperture ratio can only be acquired as discrete values between 3 and 6. We needed to acquire a higher resolution eyelid aperture ratio than 3-6, to produce a smooth and realistic facial motion with the HRP-4C robot. To solve this problem, we developed a subpixel algorithm for estimating the aperture ratio.

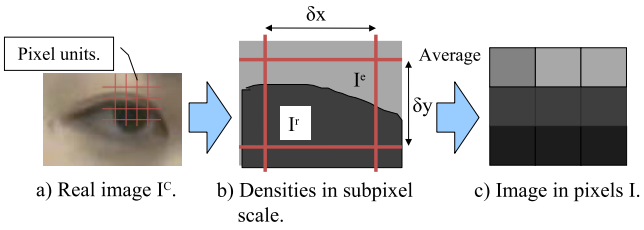


Fig. 8. Relation of pixel and real image.

In our subpixel algorithm, we assume that each observed pixel is a sum of the continuous intensity of the real object I^C:

$$I_{xy} = \frac{\int_{y-\frac{\delta y}{2}}^{y+\frac{\delta y}{2}} \int_{x-\frac{\delta x}{2}}^{x+\frac{\delta x}{2}} I^C dxdy}{\delta x \delta y}, \quad (6)$$

where δx and δy are the height and width of the pixels, respectively (see Fig.8).

If we assume that the intensity of the eyelid is I^e and the intensity of the iris is I^r, then the pixel between the boundary of the eyelid and the iris will be observed as a weighted sum of both intensities:

$$\begin{aligned} I_{xy} &= \frac{\int_{b}^{y+\frac{\delta y}{2}} \int_{x-\frac{\delta x}{2}}^{x+\frac{\delta x}{2}} I^e dxdy + \int_{y-\frac{\delta y}{2}}^{b} \int_{x-\frac{\delta x}{2}}^{x+\frac{\delta x}{2}} I^r dxdy}{\delta x \delta y} \\ &= \frac{(y + \frac{\delta y}{2} - b)I^e + (b - (y - \frac{\delta y}{2}))I^r}{\delta y}, \end{aligned} \quad (7)$$

where b is the boundary position of the eyelid.

When we accept this assumption, we can estimate the boundary position b given the pixel value I_{xy}, which has a higher resolution than δy as follows.

$$b = \frac{\delta y I_{xy} - (y + \delta y/2)I^e + (y - \delta y/2)I^r}{I^r - I^e} \quad (8)$$

I^r is given as a constant, while I^e is calculated from neighboring pixels outside of the iris.

In this specific experiment, we used a simplified formula:

$$a = \begin{cases} 0 & (e < e_{min}) \\ \frac{e-e_{min}}{e_{max}-e_{min}} & (e_{min} \le e < e_{max}) \\ 1 & (e \ge e_{max}) \end{cases} \quad (9)$$

$$e := \sum_{p_x-p_r}^{p_x+p_r} I_{xy}, \quad e_{min} := 2I^e p_r, \quad e_{max} := 2I^r p_r,$$

where p is the position of the iris (estimated in the previous step), p_r is the radius of the iris, and a is the aperture rate.

D. Mouth aperture detection

A mouth aperture is the distance between the upper and lower lips, which can be calculated from the feature points (e.g., the distance between feature points 202 and 206) as shown in Fig.6. Unfortunately, faceAPI occasionally failed to track the singer's lips in our experiment, because they moved too fast. To solve this problem, we added an extra image-processing step to detect lip movements in the video.

First, we extract a line in the image along the face center line[3], which was estimated by faceAPI. Lines obtained from the video frames are combined into a 2D image that represents the vertical motion of lips during a performance.

To estimate the center of the upper and lower lips, we used a particle filter to track lip areas using RGB color distance. Fig.9 shows the estimated lip motions.

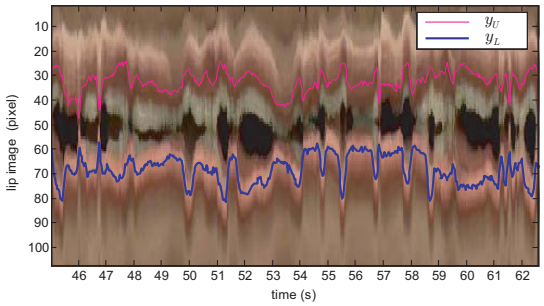


Fig. 9. Lip motion estimated by particle filter.

We define the mouth aperture ratio c as follows:

$$c = \begin{cases} 0 & (d < d_{min}) \\ \frac{d-d_{min}}{d_{max}-d_{min}} & (d_{min} \le d < d_{max}) \\ 1 & (d \ge d_{max}) \end{cases} \quad (10)$$

$$d := |y_U - y_L|,$$

where y_U and y_L are the positions of the upper and lower lip, which were obtained by the particle filter. The parameters d_{min} and d_{max} were determined, such that $c = 0$ represents the closed mouth and $c = 1$ represents the mouth at the maximum aperture.

[3]We defined the face center line to pass the feature point 202 in parallel with the vector from the point 101 to 104 in Fig.6.

The bold line in Fig.10 shows the mouth aperture at the beginning of the song. The shaded bands indicate the starts and ends of the moras obtained using VocaListener, as explained in Section III. The sound of each mora is indicated at the top. Using this graph, we observed the following features of human lip movement.

1) The mouth was open even when the singer was not pronouncing the lyrics. For example, the mouth was open for about forty percent of the period between 19.6 and 20.2 s, which was just before the start of the song. This indicates that the singer took a breath. Another breath can be observed between 22.1 and 22.5 s. In this period, the singer opened her mouth even wider than during the singing period.
2) The mouth aperture did not remain constant during each mora. Typical examples are in the /ma/ sounding period from 21 to 21.2 s and the period from 22.5 to 22.8 s. The singer almost closed her mouth at the beginning then opened it again quickly. This is a feature of the lip movements with the bilabial consonants /m/, /p/, /b/, etc. We also saw variable fluctuations in other mora periods, which were related to consonants or the emotional expression of the singer.

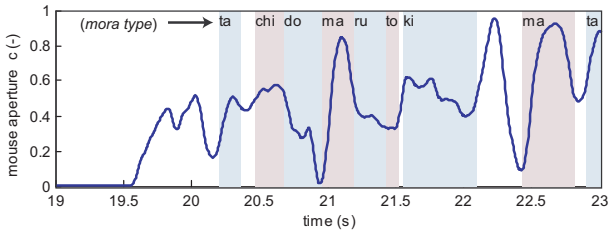


Fig. 10. Mouth aperture ratio and the mora.

V. Robot Motion Generation

Fig.11 shows the joints in the head of HRP-4C [19]. All joints were driven by servomotors, which are position-controlled and accept the target position references at 200 Hz. Therefore, our goal was to calculate a reference trajectory specifying the joint angles every 5 ms. The following subsections describe the generation process for the neck, eye and mouth references.

A. Neck motion

The robot had three neck joints, NECK_Y, NECK_P, and NECK_R that rotate the head around the yaw, pitch, and roll axes, respectively. We used the head postures obtained in IV-A with low-pass filtering as the reference trajectories for these joints.

B. Eye motion

The EYELID_P joint drives the eyelids together on both the right and left eyes. Likewise, the EYE_Y and EYE_P joints drive both eyeballs around the yaw and pitch axes, respectively.

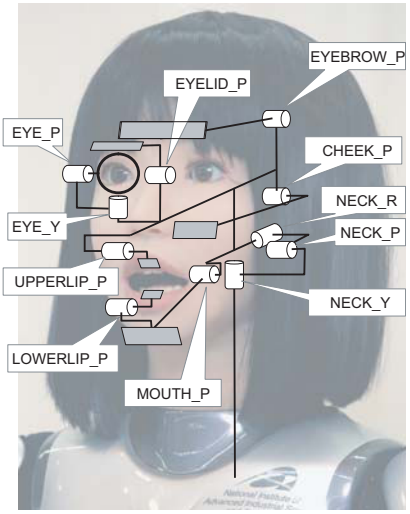


Fig. 11. Face and neck joints of HRP-4C [19].

TABLE I

JOINTS AFFECT LIP SHAPE.

Joint name	purpose	movable range (deg)
MOUTH_P	open/close the jaw	0 – 10
UPPERLIP_P	lift up/down the upper lip	-25 – 0
LOWERLIP_P	extract/retract the lower lip	0 – 25
CHEEK_P	lift up/down the mouth corners	-3.3 – 0

To reproduce the human singer's expressions, we used the eyelid aperture calculated by (9) in IV-C. The reference trajectory for EYELID_P was set as the low-pass filtered eyelid aperture multiplied by an appropriate gain.

To operate the gaze action, we specified the EYE_Y reference using the iris position obtained by (5) in IV-C.

The EYBROW_P and EYE_P joints were kept at zero; therefore, the robot maintained her eyebrows at a constant height and exhibited a horizontal gaze motion.

C. Mouth motion

Four joints were used to control the lip shape of HRP-4C, as shown in Table I. Each joint had a distinct role and a specific movable range. When we set all the joint angles to zero, the robot closed her mouth. These joint angles were calculated based on the mora obtained by VocaListener in III, and the mouth aperture obtained in IV-D. Thus, all parameters were derived from actual visual and auditory information.

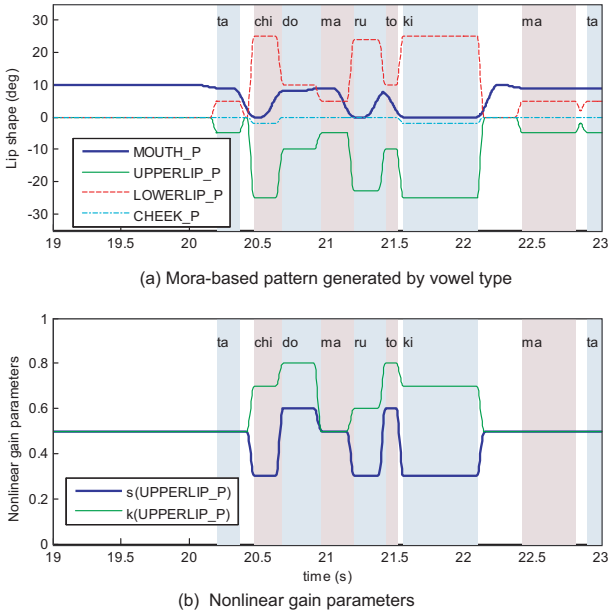
There is known to be a characteristic mouth shape for each vowel. We specified the key mouth shapes for the five vowels used in Japanese language (a,i,u,e,o) and for a breath, as shown in Table II. The nonlinear gains s and k are extra motion parameters, which will be explained later.

Using each mora type and its timing, we generated a pattern that corresponded to the vowel sequence in the song. During this step, all periods without moras were considered to be breathing. Fig.12(a) shows the pattern generated using this method. The joint trajectories were generated so they reached the key mouth shape angle at the start time of each

TABLE II

KEY POSES FOR VOWELS AND BREATH.

Vowel	/a/	/i/	/u/	/e/	/o/	Breath
MOUTH_P (deg)	9	0	0	6	8	10
UPPERLIP_P (deg)	-5	-25	-23	0	-10	0
LOWERLIP_P (deg)	5	25	24	0	10	0
CHEEK_P (deg)	0	-2	0	-1	0	0
Nonlinear gain s (-)	0.5	0.3	0.3	0.6	0.6	0.5
Nonlinear gain k (-)	0.5	0.7	0.6	0.8	0.8	0.5



(a) Mora-based pattern generated by vowel type

(b) Nonlinear gain parameters

Fig. 12. Patterns generated from the mora.

mora. We generated a smooth path using cubic splines and specified a transient time for each joint.

This *mora-based pattern* allowed our robot to achieve nominal vowel lip shapes with correct timing nevertheless, although it did not reflect the breath timing, bilabial consonants, or emotional expression of the original singer seen in the mouth aperture data of Fig.10.

A straightforward solution is to multiply the mora-based pattern by the mouth aperture ratio. In other words, the mora-based pattern is modulated by the mouth aperture. We tested this and confirmed that a natural lip motion could be generated. However, we also observed that the lip movement for the vowels /i/, /u/, and /o/ was smaller than the specified key mouth shape. This was because the aperture ratio was not normalized for each key mouth shape; instead, it was normalized for the maximum mouth opening. For example, the human mouth aperture ratio for the /i/ sound hardly exceeds 0.6; thus, the modulated lip motion is always less than sixty percent of the desired key mouth shape.

To solve this problem, we introduced a modulation using the nonlinear gain shown in Fig.13. For a given mouth aperture ratio, c, the parameters $\{s, k\}$ determine the shape of the nonlinear gain $g(c, s, k)$ required for modulation, such

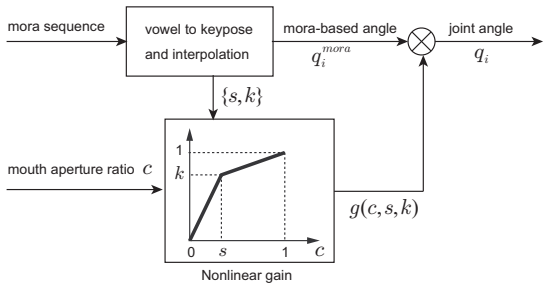


Fig. 13. Mouth aperture modulation.

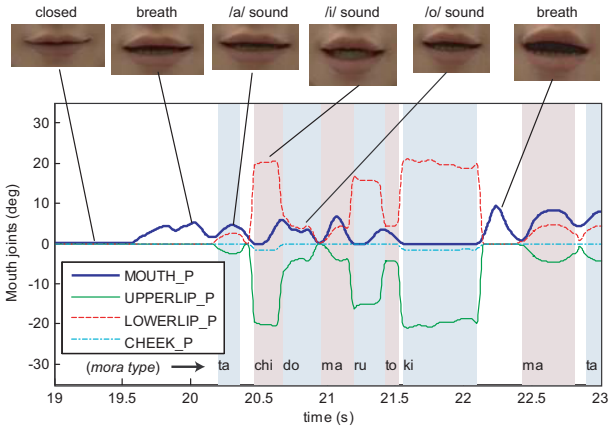


Fig. 14. Generated mouth motion.

that

$$q_i = g(c,s,k)q_i^{mora} \quad (11)$$

$$g(c,s,k) := \begin{cases} (k/s)c & (0 \le c < s) \\ \frac{1-k}{1-s}(c-s)+k & (s \le c \le 1) \end{cases}$$

where q_i and q_i^{mora} are the i-th mouth joint angles and the mora-based angle, respectively.

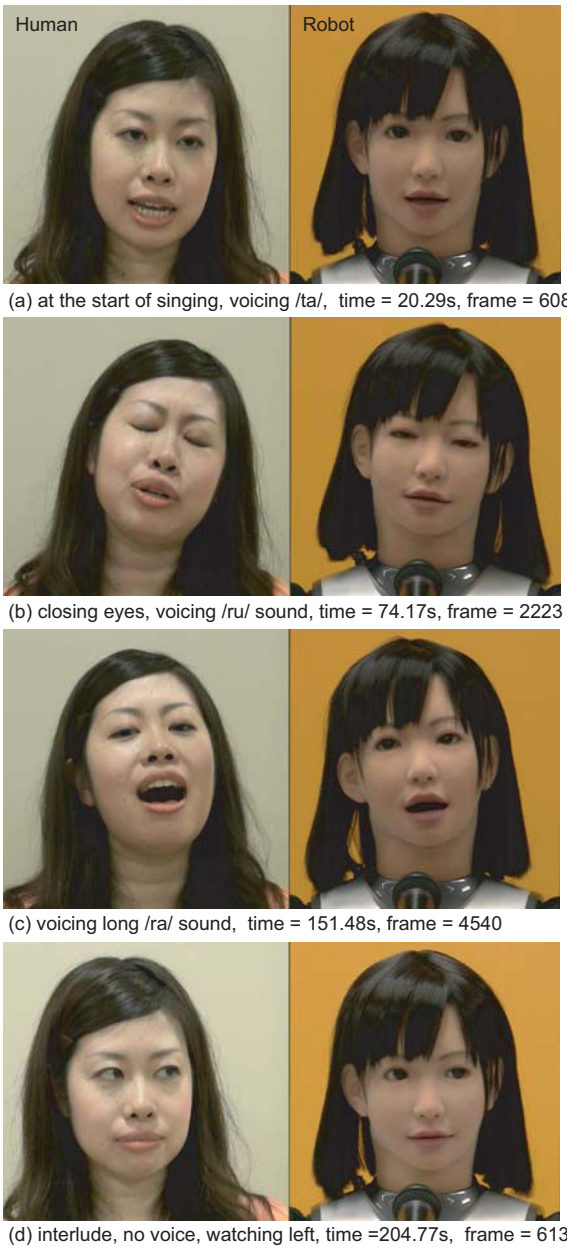
The parameters $\{s,k\}$ were determined for each vowel, as shown in the two rows at the bottom of Table II. These parameters were changed over time by a mora sequence and smoothly interpolated by cubic spline (Fig.12(b)).

Fig.14 shows the pattern generated and the some examples of lip shapes produced by HRP-4C. Our method allowed us to produce a lip sync motion that was accurate and expressive.

VI. Results

Figure 15 shows four different pairs of snapshot of the original human face (left column) and the corresponding face of the singing HRP-4C (right column). This figure shows that we succeeded in effectively making the robot imitate the original human singer. However, when we compared the human and robot faces in detail we still observed the following differences.

1) The mouth aperture of HRP-4C was smaller than that of the human singer in Fig.15(a) and (c). This was caused by a technical limitation of the current mechanism in our robot.



(a) at the start of singing, voicing /ta/, time = 20.29s, frame = 608

(b) closing eyes, voicing /ru/ sound, time = 74.17s, frame = 2223

(c) voicing long /ra/ sound, time = 151.48s, frame = 4540

(d) interlude, no voice, watching left, time =204.77s, frame = 6137

Fig. 15. Snapshots of the face of the original human singer (on the left) and the face of the robot singer (on the right) having the generated facial expressions imitating the human singer.

2) In Fig.15(b), the robot eyes were not tightly closed, unlike the human eyes. This was caused by the current eyelid joint setting, which maintains a slight open state to avoid problems such as over-current and motor burning.

One of the advantages of our robot system can be seen in the interlude of the song (Fig.15(d)). In this snapshot, the human singer voluntarily moved her head and eyes even though she was not singing, and these movements were well imitated by our robot. We found that the imitation of such subtle and unconscious motions contributed to making our robot performance appear much more realistic.

Fig. 16. Our robot singer HRP-4C on the stage in CEATEC JAPAN 2010.

We demonstrated HRP-4C singing at CEATEC JAPAN 2010. This is Japan's largest consumer electronics exhibition and it was held in Chiba city in October 2010. To make the robot performance more attractive, we choreographed arm motions so that the arms could move naturally in time to the music. This was achieved using our own motion creation software, *Choreonoid*[20]. We did not include leg motions, because the robot was fixed on a stand to perform songs in a standing posture. The demonstration of our singing robot was successful and it received a lot of audience and media coverage.[4] We found that some of the audience did not realize that the singer was a robot at first sight, because of the naturalness of the singing voice and motions. However, other audience members also expressed a feeling of 'creepiness'. Demonstration video clips are available on our website (http://staff.aist.go.jp/t.nakano/VocaWatcher/).

VII. Conclusions and Future Work

We have described a humanoid robot singer that sings with highly natural synthesized singing voices and that produces highly realistic facial expressions. By integrating visual and audio information, the robot motions synchronized with singing voices were generated from a human singer performance recorded using a single camera.

We aim to improve the robot mechanism and overcome some current limitations in the singing motions. For example, HRP-4C cannot open her mouth sufficiently wide and she cannot close her eyes tightly. Furthermore, it takes a day to generate robot motions from a given a video clip in the current implementation, which needs to be reduced in future. Our experiences have shown us that a realistic robot singer attracts considerable public attention, and we hope to make it more widely available in future.

[4]Some examples are `http://www.diginfo.tv/2010/10/13/10-0217-r-en.php` and `http://blogs.wsj.com/japanrealtime/2010/10/05/japans-next-pop-idol-is-a-robot/` .

Acknowledgements

We thank the members of the Humanoid Robotics Group of AIST, especially Kanako Miura for her help and advice at an early stage of the experiment and Kenta Yonekura for his work choreographing HRP-4C's arm motions. We also thank Yoshio Matsumoto for his helpful advice. Finally, we gratefully acknowledge the support of Hirohisa Hirukawa and Satoshi Sekiguchi, the directors of the Intelligent Systems Research Institute and the Information Technology Research Institute of AIST, respectively.

References

[1] I. Kato *et al.*, "The robot musician WABOT-2," *Robotics*, vol.3, pp.143-155, 1987.

[2] K. Chida *et al.* "Development of a new anthropomorphic flutist robot WF-4," in *Proc. of ICRA2004*, pp.152-157, 2004.

[3] T.Mizumoto *et al.* "Thereminist Robot: Development of a robot theremin player with feedforward and feedback arm control based on a theremin's pitch model," in *Proc. of IROS2009*, 2009.

[4] Y. Kuroki, M. Fujita, T. Ishida, K. Nagasaka and J. Yamaguchi, "A small biped entertainment robot exploring attractive applications," in *Proc. of ICRA2003*, pp.471-476, 2003.

[5] K. Murata, K. Nakadai, *et al.*, "A robot singer with music recognition based on real-time beat tracking," in *Proc. of ISMIR 2008 – Session 2b – Music Recognition and Visualization*, pp.199-204, 2008.

[6] K.Kaneko, F.Kanehiro, M.Morisawa, K.Miura, S.Nakaoka and S.Kajita "Cybernetic Human HRP-4C," in *Proc. IEEE/RSJ Int. Conference on Humanoid Robots*, pp.7-14, 2009.

[7] M.Tachibana, S.Nakaoka and H.Kenmochi, "A singing robot realized by a collaboration of VOCALOID and Cybernetic Human HRP-4C," in *Proc. of InterSinging2010*, Tokyo, 2010.

[8] F. Wilbers, C. Ishi and H. Ishiguro, "A blendshape model for mapping facial motions to an android," in *Proc. of the IROS2007*, pp.542-547, 2007.

[9] P. Jaeckel, N. Campbell, C. Melhuish, "Facial behavior mapping — From video footage to a robot head, " *Robotics and Autonomous Systems*, vol.56, pp.1042-1049, 2008.

[10] M. Goto *et al.*, "RWC music database: Music genre database and musical instrument sound database," in *Proc. of ISMIR2003*, pp.229-230, 2003.

[11] T.Nakano and M.Goto, "VocaListener: A singing-to-singing synthesis system based on iterative parameter estimation," in *Proc. of the SMC 2009*, pp.343-348, 2009.

[12] H. Kenmochi and H. Ohshita, "Vocaloid – Commercial singing synthesizer based on sample concatenation," in *Proc. Interspeech 2007*, pp. 4010-4011, 2007.

[13] J. Janer, J. Bonada, and M. Blaauw, "Performance-driven control for sample-based singing voice synthesis," in *Proc. of DAFx-06*, pp.41-44, 2006.

[14] T. Nakano, J. Ogata, M. Goto, and Y. Hiraga, "Analysis and automatic detection of breath sounds in unaccompanied singing voice," in *Proc. ICMPC 10*, 2008.

[15] H. Kawahara, M. Morise, T. Takahashi, R. Nisimura, T. Irino, and H. Banno, "Tandem-STRAIGHT: A temporally stable power spectral representation for periodic signals and applications to interference-free spectrum, F0, and aperiodicity estimation," in *Proc. of ICASSP 2008*, pp. 3933-3936, 2008.

[16] Seeing Machines `http://www.seeingmachines.com`

[17] Y. Matsumoto, T. Ino and T. Ogasawara, "Development of intelligent wheelchair system with face and gaze based interface," in *Proc. of ROMAN 2001*, pp.262-267, 2001.

[18] T. Morris, P. Blenkhorn and F. Zaidi, "Blink detection for real-time eye tracking," *Journal of Network and Computer Applications*, Volume 25, Issue 2, pp.129-143, 2002.

[19] S. Nakaoka, F. Kanehiro, K. Miura, *et al.* "Creating facial motions of Cybernetic Human HRP-4C," in *Proc. of Humanoids2009*, pp.561-567, 2009.

[20] S. Nakaoka, S. Kajita and K. Yokoi, "Intuitive and flexible user interface for creating whole body motions of biped humanoid robots," in *Proc. of the IROS2010*, pp.1675-1682, 2010.