

SURAT - TUGAS

Nomor: 853-D/3485/FT-UNTAR/VIII/2020

Dekan Fakultas Teknik Universitas Tarumanagara, dengan ini menugaskan kepada Saudara :

Dr. Hugeng, S.T., M.T.

Untuk pencatatan **Ciptaan/Hak Cipta** dengan data sebagai berikut :

Judul : Program Program Rekonstruksi Head Related Impulse Responses (HRIR) Menggunakan Pemodelan Principal Components Analysis (PCA) Dan Interpolasi Bilinier Segiempat Untuk Menghasilkan Audio Tiga Dimensi Pada Posisi Yang Dikehendaki
Jenis : Program Komputer
Nomor : EC00201946199
Jangka Waktu : 50 (lima puluh) tahun

Demikian Surat Tugas ini dibuat, untuk dilaksanakan dengan sebaik-baiknya dan melaporkan hasil penugasan tersebut kepada Dekan Fakultas Teknik Universitas Tarumanagara.

24 Agustus 2020

Dekan


Harto Tanujaya, S.T., M.T., Ph.D.

Tembusan:

1. Kaprodi. Sarjana Teknik Elektro
2. Kasubag. Personalia
3. Arsip

PROGRAM STUDI :

- Sarjana Arsitektur, Magister Arsitektur, Sarjana Perencanaan Wilayah dan Kota, Magister Perencanaan Wilayah dan Kota
- Sarjana Teknik Sipil, Magister Teknik Sipil, Doktor Teknik Sipil
- Sarjana Teknik Mesin, Sarjana Teknik Industri, Sarjana Teknik Elektro

Jl. Letjen. S. Parman No.1 - Jakarta 11440

P : (021) 5663124 - 5672548 - 5638335

MPWK : (021) 56967322, MTS : (021) 5655801 - 5655802, DTS : (021) 56967015 - 5645907

F : (021) 5663277, MTS : (021) 5655805, MPWK : (021) 5645956

E : ft@untar.ac.id

www.untar.ac.id



REPUBLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN CIPTAAN

Dalam rangka pelindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan

: EC00201946199, 16 Juli 2019

Pencipta

Nama

: Dr. Hugeng, S.T., M.T., Jovan Anggara,, S.Kom., , dkk

Alamat

: Pesing Garden No. 120 RT.001 RW.008 Kedoya Utara, Kebon Jeruk, Jakarta Barat, Dki Jakarta, 11520

Kewarganegaraan

: Indonesia

Pemegang Hak Cipta

Nama

: Dr. Hugeng, S.T., M.T., Jovan Anggara, S.Kom., , dkk

Alamat

: Pesing Garden No. 120 RT.001 RW.008 Kedoya Utara, Kebon Jeruk, Jakarta Barat, 7, 11520

Kewarganegaraan

: Indonesia

Jenis Ciptaan

: Program Komputer

Judul Ciptaan

: Program Program Rekonstruksi Head Related Impulse Responses (HRIR) Menggunakan Pemodelan Principal Components Analysis (PCA) Dan Interpolasi Bilinier Segiempat Untuk Menghasilkan Audio Tiga Dimensi Pada Posisi Yang Dikehendaki

Tanggal dan tempat diumumkan untuk pertama kali di wilayah Indonesia atau di luar wilayah Indonesia

: 15 Juli 2019, di Jakarta

Jangka waktu pelindungan

: Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.

Nomor pencatatan

: 000146365

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.
Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.



a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL

Dr. Freddy Harris, S.H., LL.M., ACCS.
NIP. 196611181994031001

LAMPIRAN PENCIPTA

No	Nama	Alamat
1	Dr. Hugeng, S.T., M.T.	Pesing Garden No. 120 RT.001 RW.008 Kedoya Utara, Kebon Jeruk
2	Jovan Anggara,, S.Kom.	Perum. Banjar Wijaya, Cluster Cattleya Blok B60/9
3	Tommy Sutiawan, S.Kom.	Jl. Gama 4 No.15, Cimone Permai
4	Irfan Naufal, S.Kom.	Jl. Zaitun 2 No.9 Blok B2, Islamic Village, Kelapa Dua

LAMPIRAN PEMEGANG

No	Nama	Alamat
1	Dr. Hugeng, S.T., M.T.	Pesing Garden No. 120 RT.001 RW.008 Kedoya Utara, Kebon Jeruk
2	Jovan Anggara, S.Kom.	Perum. Banjar Wijaya, Cluster Cattleya Blok B60/9
3	Tommy Sutiawan, S.Kom.	Jl. Gama 4 No.15, Cimone Permai
4	Irfan Naufal, S.Kom.	Jl. Zaitun 2 No.9 Blok B2, Islamic Village, Kelapa Dua

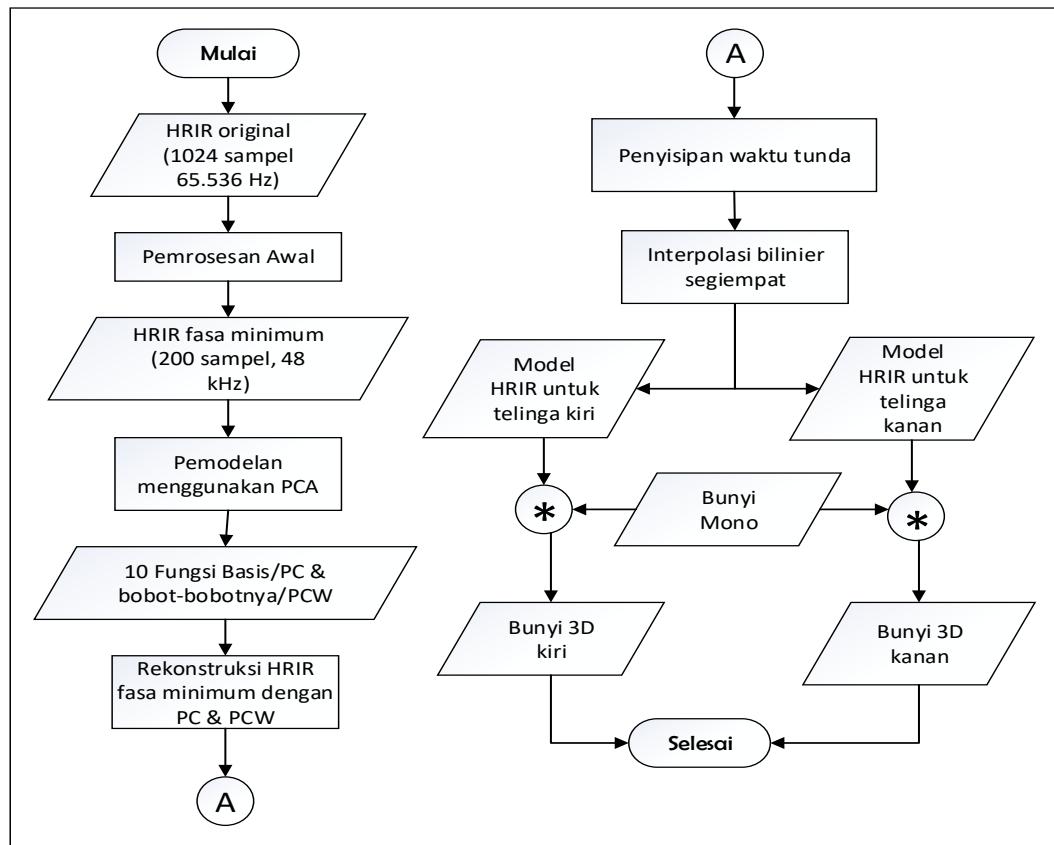


PROGRAM

“REKONSTRUKSI HEAD RELATED IMPULSE RESPONSES (HRIR) MENGGUNAKAN PEMODELAN PRINCIPAL COMPONENTS ANALYSIS (PCA) DAN INTERPOLASI BILINIER SEGIEMPAT UNTUK MENGHASILKAN AUDIO TIGA DIMENSI PADA POSISI YANG DIKEHENDAKI”

Tahap awal dalam pembuatan program ini adalah pemilihan basis data *Head Related Impulse responses* (HRIR) yang sesuai. Di sini digunakan basis data PKU-IOA HRTF Database (Qu et al., 2009). Keunggulan dari PKU-IOA HRTF Database adalah bahwa data HRIR dicuplik dari sumber-sumber bunyi pada permukaan bola dengan jarak-jarak yang berbeda, yaitu 20 – 160 cm dari pusat kepala subyek. Tetapi data HRIR diambil hanya dari satu manekin *Knowles Electronic Manikin for Acoustic Research* (KEMAR).

Program rekonstruksi HRIR yang diciptakan ini dibuat dengan menggunakan piranti lunak bantu Matlab. Data HRIR yang digunakan berasal dari pencuplikan semua sumber bunyi pada permukaan bola dengan jarak 100 cm dari pusat kepala KEMAR, yaitu masing-masing sejumlah 793 HRIR untuk satu telinga. Gambar 1 menunjukkan diagram alir program yang diciptakan untuk rekonstruksi HRIR menggunakan *PKU-IOA HRTF Database*.



Gambar 1. Diagram Alir Program Rekontruksi HRIR Menggunakan *PKU-IOA HRTF Database*

Program rekonstruksi HRIR ini dibuat dalam satu program utama (*main script*), seperti ditunjukkan Gambar 2, yang menjalankan program-program sesuai dengan proses-proses yang terjadi dalam diagram alir pada Gambar 1. Program dimulai dengan menentukan data HRIR mana yang akan digunakan. Di sini digunakan data HRIR yang diambil dari semua posisi pada permukaan bola dengan jarak 100 cm dari pusat kepala KEMAR. Kemudian setiap HRIR diubah formatnya dari .dat (format original dari PKU-IOA HRTF Database) menjadi .mat, yaitu format data Matlab. Data HRIR disimpan dalam nama file dan folder yang sesuai. Gambar 3 menunjukkan Program Pemrosesan Awal dari diagram alir pada Gambar 1.

```
run_hrir_mph_pca_rect_all.m  ×  make_hrir_dat_d  
1 -      get_HRTF_coords;  
2 -      make_hrir_dat_dist100_mph_all;  
3 -      pca_hrir256_mph_dist100_all  
4 -      make_estim_hrir256_dist100  
5 -      make_hrir_interp_rect;
```

Gambar 2. Program Utama Rekonstruksi HRIR Menggunakan PCA dan Interpolasi

```
1 - clear all;
2 - load coordinates.mat
3 - fid=fopen('hrir_dat_dist100_mph_all.m','wt');
4 -
5 - for i = 1:length(aziEleDist);
6 -     koord = squeeze(aziEleDist(i,:));
7 -     fprintf(fid,'koord(%s) = squeeze(aziEleDist(%s,:));\n',i,i);
8 -     fprintf(fid,'hl = readhrir(''1'',%d,%d,%d);\n',koord(3),koord(2),koord(1));
9 -     fprintf(fid,'hr = readhrir(''r'',%d,%d,%d);\n',koord(3),koord(2),koord(1));
10 -    fprintf(fid,'hrir_l = hl(); hrir_r = hr();\n');
11 -    fprintf(fid,'save dist%d\lev%d\azi%d_elev%d_dist%d.mat hrir_l hrir_r;\n',koord(3),koord(2),koord(1),koord(2),koord(3));
12 -    fprintf(fid,'hl = resample(hrir_l, 48000, 65536); hrir_l = hl;\n');
13 -    fprintf(fid,'hr = resample(hrir_r, 48000, 65536); hrir_r = hr;\n');
14 -    fprintf(fid,'save dist%d\lev%d\azi%d_elev%d_48kHz_dist%d.mat hrir_l hrir_r;\n',koord(3),koord(2),koord(1),koord(2),koord(3));
15 -    fprintf(fid,'fprintf(''azi%d_elev%d_dist%d.mat DONE \n'');\n',koord(1),koord(2),koord(3));
16 -    fprintf(fid,['zz,hlm] = rceps(hrir_l);\n');
17 -    fprintf(fid,['zz,hrm] = rceps(hrir_r);\n');
18 -    fprintf(fid,'hrir_mph_l = hlm(1:256); hrir_mph_r = hrm(1:256);\n');
19 -    fprintf(fid,'hrir_all(%d,:)=hrir_mph_l';\n',i);
20 -    fprintf(fid,'hrir_all(%d+length(aziEleDist),:)=hrir_mph_r';\n',i);
21 -    fprintf(fid,'save dist%d\lev%d\azi%d_elev%d_dist%d_mph256.mat hrir_mph_l hrir_mph_r;\n',koord(3),koord(2),koord(1),koord(2),koord(3));
22 -    fprintf(fid,'fprintf(''azi%d_elev%d_dist%d_mph256.mat DONE \n'');\n',koord(1),koord(2),koord(3));
23 - end
24 - fprintf(fid,'save hrir256_mph_dist100_all.mat hrir_all;\n');
25 - fclose(fid);
26 - hrir_dat_dist100_mph_all
```

Gambar 3. Program Pemrosesan Awal Untuk Rekonstruksi HRIR Menggunakan PCA dan Interpolasi

Di sini terdapat total 1586 HRIR, dimana masing-masing HRIR terdiri dari 1024 sampel dengan frekuensi sampling sebesar 65.536 Hz. Selanjutnya frekuensi sampling dari semua HRIR original

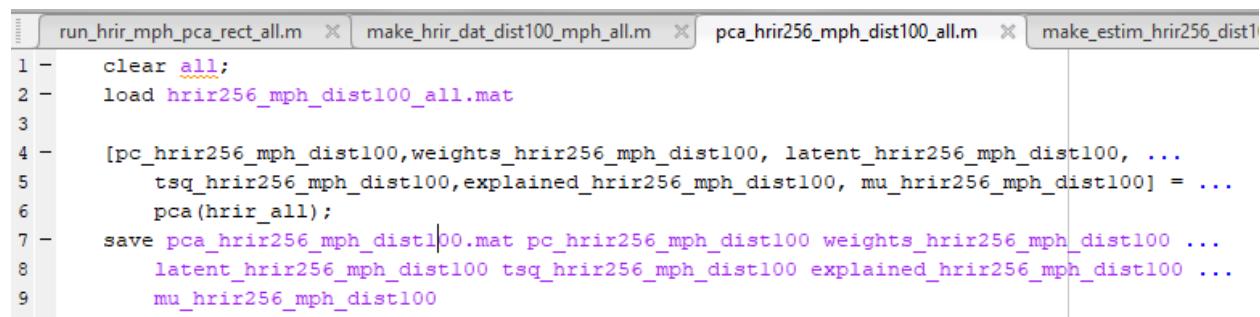
dikonversi menjadi 48 kHz karena frekuensi sampling ini banyak digunakan dengan alasan bahwa frekuensi sampling ini memenuhi syarat Nyquist, yaitu bahwa diperlukan frekuensi sampling sebesar minimal dua kali dari frekuensi maksimal yang dimiliki suatu sinyal. Seperti diketahui bahwa frekuensi audio maksimal yang dapat didengar manusia adalah 20 kHz. Fungsi `resample` digunakan untuk konversi HRIR dari 65.536 Hz menjadi 48.000 Hz, sehingga jumlah sampel HRIR berubah dari 1024 menjadi 750.

Tahap selanjutnya pada Pemrosesan Awal adalah mengubah semua HRIR yang memiliki frekuensi sampling 48 kHz tadi, menjadi HRIR yang memiliki fasa minimum. Dari beberapa penelitian, diperoleh bahwa fungsi filter telinga manusia (HRIR) dapat didekati dengan respons impuls dengan fasa minimum. HRIR fasa minimum adalah versi HRIR dengan waktu tunda yang dihilangkan yang memiliki fasa minimum. Jumlah sampel HRIR fasa minimum yang diambil hanya 256 sampel pertama karena sampel-sampel berikutnya memiliki amplitudo sangat kecil sehingga dapat diabaikan. Untuk memudahkan proses pemodelan menggunakan PCA, semua HRIR fasa minimum dikumpulkan dalam satu matriks `hrir_all` yang berukuran 1586 x 256, dimana 793 baris pertama dari matriks adalah HRIR fasa minimum untuk telinga kiri untuk semua posisi sumber bunyi dan 793 baris berikutnya untuk telinga kanan.

Pemodelan *Principal Components Analysis* (PCA) dilakukan terhadap seluruh data HRIR (`hrir_all`), seperti terlihat pada Gambar 4. Pemodelan PCA mengurangi dimensi dari data yaitu semula berdimensi 256 (sesuai dengan jumlah sampel dalam satu HRIR), menjadi berdimensi 10. Dipilih dimensi 10 karena digunakan hanya 10 fungsi basis (PC), dari total 256 fungsi basis, yang mewakili lebih dari 90% variasi yang terdapat dalam data HRIR total. Pemodelan PCA menggunakan 10 fungsi basis tepatnya mewakili 91,91% dari keseluruhan data HRIR. Model HRIR fasa minimum yang ke- i , dimana $i = 1, 2, 3, \dots, 1586$, dapat dituliskan dalam persamaan berikut:

$$\text{hrir_mph}_i = \text{PCW}_{1,i} \text{PC}_1 + \text{PCW}_{2,i} \text{PC}_2 + \text{PCW}_{3,i} \text{PC}_3 + \dots + \text{PCW}_{10,i} \text{PC}_{10} \quad (1)$$

dimana $\text{PC}_1, \text{PC}_2, \dots, \text{PC}_{10}$ adalah fungsi basis ke-1 sampai ke-10 dengan jumlah sampel masing-masing PC sebesar 256, $\text{PCW}_{n,i}$ adalah bobot PC_n untuk model HRIR fasa minimum yang ke- i , hrir_mph_i . Pemodelan PCA menggunakan hanya 10 fungsi basis untuk HRIR fasa minimum keseluruhan dapat dilihat pada Gambar 5.



```

1 - clear all;
2 - load hrir256_mph_dist100_all.mat
3
4 - [pc_hrir256_mph_dist100,weights_hrir256_mph_dist100, latent_hrir256_mph_dist100, ...
5 -     tsq_hrir256_mph_dist100,explained_hrir256_mph_dist100, mu_hrir256_mph_dist100] = ...
6 -     pca(hrir_all);
7 - save pca_hrir256_mph_dist100.mat pc_hrir256_mph_dist100 weights_hrir256_mph_dist100 ...
8 -     latent_hrir256_mph_dist100 tsq_hrir256_mph_dist100 explained_hrir256_mph_dist100 ...
9 -     mu_hrir256_mph_dist100

```

Gambar 4. Pemodelan PCA dari HRIR fasa minimum

Setelah diperoleh model HRIR fasa minimum dari Persamaan 1, diperlukan waktu tunda yang berkaitan dengan jarak yang ditempuh bunyi dari posisi tertentu ke masing-masing gendang telinga pendengar. Waktu tunda ini disisipkan pada model HRIR fasa minimum untuk rekonstruksi HRIR kedua telinga.



```

1 - clear all;
2 - load coordinates.mat
3 - load pca_hrir256_mph_dist100.mat
4 - fid=fopen('estim_hrir256_dist100.m','wt');
5
6 - for i = 1:NumHRTFs_perDist;
7 -   koord = squeeze(aziEleDist(i,:));
8 -   fprintf(fid,'estim_hl256_mph = (weights_hrir256_mph_dist100(:,1:10)*pc_hrir256_mph_dist100(:,1:10)')+mu_hrir256_mph_dist100;\n',i);
9 -   fprintf(fid,'estim_hr256_mph = (weights_hrir256_mph_dist100(:,1:10)*pc_hrir256_mph_dist100(:,1:10)')+mu_hrir256_mph_dist100;\n',i+NumHRTFs_perDist);
10 -  fprintf(fid,'save dist%d\alev%d\estim_hrir256_mph_azi%d_elev%d_dist%d.mat estim_hl256_mph estim_hr256_mph;\n',koord(3),koord(2),koord(1),koord(2),koord(3));
11 -  fprintf(fid,'fprintf(''estim_hrir256_mph_azi%d_elev%d_dist%d.mat DONE \n'');\n\n',koord(1),koord(2),koord(3));
12 -  fprintf(fid,'dL = sqrt(100^2 + 7.5^2 - 2*100*7.5*cos((90+d)*pi/180));\n', koord(1)); % distance of sound source to left ear in cm
13 -  fprintf(fid,'dR = sqrt(100^2 + 7.5^2 - 2*100*7.5*cos((90-d)*pi/180));\n', koord(1)); % distance of sound source to right ear in cm
14 -  fprintf(fid,'timD_l(%d) = dL / 34.3;\n', i, i); % time delay in ms
15 -  fprintf(fid,'OnL(%d) = round(timD_l(%d)*48);\n',i,i); % number of samples according to time delay timD_l, fs=48 kHz
16 -  fprintf(fid,'OnR(%d) = round(timD_r(%d)*48);\n',i,i); % number of samples according to time delay timD_r, fs=48 kHz
17 -  fprintf(fid,'estim_hl256 = [zeros(1,OnL(%d)),estim_hl256_mph(1:length(estim_hl256_mph)-OnL(%d))];\n', i,i);
18 -  fprintf(fid,'estim_hr256 = [zeros(1,OnR(%d)),estim_hr256_mph(1:length(estim_hr256_mph)-OnR(%d))];\n', i,i);
19 -  fprintf(fid,'save dist%d\alev%d\estim_hrir256_azi%d_elev%d_dist%d.mat estim_hl256 estim_hr256;\n',koord(3),koord(2),koord(1),koord(2),koord(3));
20 -  fprintf(fid,'fprintf(''estim_hrir256_azi%d_elev%d_dist%d.mat DONE \n'');\n\n',koord(1),koord(2),koord(3));
21 - end;
22 - fprintf(fid,'save time_delay_insert.mat OnL OnR timD_l timD_r;\n');
23 - fclose(fid)
24 - estim_hrir256_dist100

```

Gambar 5. Rekonstruksi HRIR dari Penyisipan Waktu Tunda pada Model HRIR Fasa Minimum

Untuk mendapatkan HRIR pada posisi yang dikehendaki dimana tidak ada data HRIR sebelumnya, digunakan interpolasi bilinier segiempat dari 4 HRIR tetangga hasil rekonstruksi di atas, seperti terlihat pada Gambar 6. Sedangkan Gambar 7 menunjukkan program yang dibuat untuk mengimplementasikan interpolasi bilinier segiempat pada Gambar 6. Sepasang HRIR hasil interpolasi dikonvolusi masing-masing dengan sebuah bunyi mono sehingga dihasilkan bunyi 3D yang dipersepsi masing-masing telinga.

Interpolasi bilinier adalah suatu interpolasi yang menggunakan satu set titik-titik tetangga pada bidang vertikal atau horizontal dari titik target yang dicari. Dalam interpolasi bilinier segiempat, 4 titik digunakan sebagai acuan. Keempat titik tersebut membentuk suatu segiempat. Perhitungan untuk interpolasi bilinier segiempat mengikuti persamaan berikut:

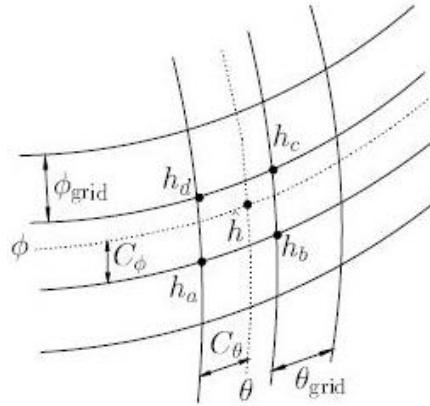
$$\hat{h} = (1 - C_\theta)(1 - C_\phi)h_a + C_\theta(1 - C_\phi)h_b + C_\theta C_\phi h_c + (1 - C_\theta)C_\phi h_d \quad (2)$$

dimana θ = sudut azimut dan ϕ = sudut elevasi, C_θ dan C_ϕ mengacu pada Persamaan 3 dan Persamaan 4 berikut ini (Freeland et al., 2004),

$$C_\theta = \frac{c_\theta}{\theta_{grid}} = \frac{\theta \text{ mod}(\theta_{grid})}{\theta_{grid}} \quad (3)$$

$$C_\phi = \frac{c_\phi}{\phi_{grid}} = \frac{\phi \text{ mod}(\phi_{grid})}{\phi_{grid}}. \quad (4)$$

Dalam Gambar 6, h_a , h_b , h_c , dan h_d adalah empat HRIR tetangga dari HRIR target, \hat{h} , yang tersedia dari basis data. Empat HRIR ini digunakan sebagai titik-titik acuan untuk kalkulasi interpolasi. Gambar 6 juga menjelaskan bagaimana memperoleh nilai-nilai C_θ and C_ϕ .



Gambar 6. Interpretasi Grafik dari Interpolasi Bilinier Segiempat (Freeland et al., 2004)

```

1 - clear all;
2 - Elevations = -40:10:90; % Elevations measured
3 - Elevations2 = -30:10:80;
4 - AzimuthStep = [repmat(5, 1, 10), 10, 15, 30, 360]; % Azimuth steps from lowest to highest elevation
5 - Distances = [100]; % Distances measured
6
7 - %azimuth and elevation
8 - for dist = Distances
9 -     elei = 1;
10 -    for ele = Elevations2
11 -        elei = elei+1;
12 -        for azi = 0:AzimuthStep(elei):359
13 -            azi0 = mod(azi - 5, 360);
14 -            azi1 = mod(azi + 5, 360);
15 -            ele0 = Elevations(elei-1);
16 -            ele1 = Elevations(elei+1);
17 -            cazi = (azi-azi0) / (azi1-azi0);
18 -            cele = (ele1-ele0) / (ele1-ele0);
19 -            load_hrir = sprintf('load dist%d\\elev%d\\estim_hrir256_azi%d_elev%d_dist%d.mat',dist,ele0,azi0,ele0,dist);
20 -            fail = false;
21 -            try
22 -                eval(load_hrir);
23 -            catch
24 -                sprintf('failed load dist%d\\elev%d\\estim_hrir256_azi%d_elev%d_dist%d.mat\n',dist,ele0,azi0,ele0,dist);
25 -                fail = true;
26 -            end
27 -            if fail == true
28 -                sprintf('dist%d elev%d hrir_azi%d_elev%d_dist%d_interpret.mat SKIPPED \n',dist,ele,azi,ele,dist);
29 -                continue;
30 -            end
31 -            a_l = (1-cazi)*(1-cele)*estim_hl256;
32 -            a_r = (1-cazi)*(1-cele)*estim_hr256;
33
34 -            load_hrir = sprintf('load dist%d\\elev%d\\estim_hrir256_azi%d_elev%d_dist%d.mat',dist,ele0,azi1,ele0,dist);
35 -            try
36 -                eval(load_hrir);
37 -            catch
38 -                sprintf('failed load dist%d\\elev%d\\estim_hrir256_azi%d_elev%d_dist%d.mat\n',dist,ele0,azi1,ele0,dist);
39 -                fail = true;
40 -            end
41 -            if fail == true
42 -                sprintf('dist%d elev%d hrir_azi%d_elev%d_dist%d_interpret.mat SKIPPED \n',dist,ele,azi,ele,dist);
43 -                continue;
44 -            end
45 -            b_l = (cazi)*(1-cele)*estim_hl256;
46 -            b_r = (cazi)*(1-cele)*estim_hr256;

```

Gambar 7. Program Interpolasi Bilinier Segiempat dari HRIR pada Posisi yang Dikehendaki

```

47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85

```

```

load_hrir = sprintf('load dist%d\\elev%d\\estim_hrir256_azi%d_elev%d_dist%d.mat',dist,elel,azil,elel,dist);
try
    eval(load_hrir);
catch
    fprintf('failed load dist%d\\elev%d\\estim_hrir256_azi%d_elev%d_dist%d.mat\n',dist,elel,azil,elel,dist);
    fail = true;
end
if fail == true
    fprintf('dist%d elev%d hrir_azi%d_elev%d_dist%d_interprect.mat SKIPPED \n',dist,ele,azi,ele,dist);
    continue;
end
c_l = (cazi)*(cele)*estim_hl256;
c_r = (cazi)*(cele)*estim_hr256;

load_hrir = sprintf('load dist%d\\elev%d\\estim_hrir256_azi%d_elev%d_dist%d.mat',dist,elel,azil0,elel,dist);
try
    eval(load_hrir);
catch
    fprintf('failed load dist%d\\elev%d\\estim_hrir256_azi%d_elev%d_dist%d.mat\n',dist,elel,azil0,elel,dist);
    fail = true;
end
if fail == true
    fprintf('dist%d elev%d hrir_azi%d_elev%d_dist%d_interprect.mat SKIPPED \n',dist,ele,azi,ele,dist);
    continue;
end
d_l = (1-cazi)*(cele)*estim_hl256;
d_r = (1-cazi)*(cele)*estim_hr256;

interp_hrir_l = a_l + b_l + c_l + d_l;
interp_hrir_r = a_r + b_r + c_r + d_r;
if fail == false
    save_hrir = sprintf('save dist%d\\elev%d\\hrir_azi%d_elev%d_dist%d_interprect.mat interp_hrir_l interp_hrir_r',dist,ele,azi,ele,dist);
    eval(save_hrir);
end
fprintf('dist%d elev%d hrir_azi%d_elev%d_dist%d.mat DONE \n',dist,ele,azi,ele,dist);
end
end

```

Gambar 7. Program Interpolasi Bilinier Segiempat dari HRIR pada Posisi yang Dikehendaki (Lanjutan)

Untuk melakukan simulasi bunyi 3D yang bergerak, di dalam Matlab dijalankan program *simulation_moving_virtual3Ds*ound. Dalam contoh di sini disimulasikan bunyi 3D pada bidang horisontal (sudut elevasi 0°) dengan 5 sudut azimuth sebesar 270°, 315°, 0°, 45°, dan 90°. Hasil simulasi adalah file audio 3D dengan nama ‘moving3Dsoun.d.wav’. Dengan membuka *file* moving3Dsoun.d.wav menggunakan aplikasi *audio player* pada komputer, maka akan terdengar bunyi helikopter stereo yang bergerak pada bidang horisontal sesuai dengan simulasi.

Referensi

- Freeland, F. P., Biscainho, L.W.P., Diniz, P.S., (2004), Interpositional transfer function of 3D-sound generation, Journal of Audio Eng. Soc, Volume 52, pp. 915-930.
- Qu, T., Xiao, Z., Gong, M., Huang, Y., Li, X., Wu, X., (2009), Distance-dependent head-related transfer functions measured with high spatial resolution using a spark gap, IEEE Trans. on Audio, Speech, and Language Processing, vol. 17, no. 6, pp. 1124-1132.

MANUAL BOOK PENGGUNAAN
PROGRAM

“REKONSTRUKSI HEAD RELATED IMPULSE RESPONSES (HRIR) MENGGUNAKAN
PEMODELAN PRINCIPAL COMPONENTS ANALYSIS (PCA) DAN INTERPOLASI
BILINIER SEGIEMPAT UNTUK MENGHASILKAN AUDIO TIGA DIMENSI PADA POSISI
YANG DIKEHENDAKI”

Dibuat Oleh:
Dr. Hugeng, S.T., M.T.
Jovan Anggara, S.Kom.
Tommy Sutiawan, S.Kom.
Irfan Naufal, S.Kom.

JAKARTA
JULI 2019

Cara penggunaan program rekonstruksi HRIR ini adalah sebagai berikut:

1. Unduh basis data PKU-IOA HRTF Database dari <http://www.cis.pku.edu.cn/auditory/Staff/Dr.Qu.files/Qu-HRTF-Database.html> (Qu et al., 2009);
2. Ekstrak basis data tersebut dalam suatu *folder* misalnya \PKU-IOA HRTF Database\;
3. Lakukan *Set Path with subfolders* pada Matlab untuk *folder* instalasi di atas;
4. Salin *file* program utama *run_hrir_mph_pca_rect_all.m* pada *folder* instalasi di atas;
5. Salin *file* berikut ke dalam folder instalasi yang sama:

```
get_HRTF_coords.m  
make_hrir_dat_dist100_mph_all.m  
pca_hrir256_mph_dist100_all.m  
make_estim_hrir256_dist100.m  
make_hrir_interp_rect.m
```

6. Jalankan program utama *run_hrir_mph_pca_rect_all* dari *command window* Matlab atau dari Editor Matlab. Setelah program utama selesai dieksekusi, maka HRIR hasil rekonstruksi telah tersedia di dalam subfolder \PKU-IOA HRTF Database\dist100\ yang bersangkutan;
7. Untuk menghasilkan bunyi 3D yang diinginkan, salin *file* audio mono ke dalam folder instalasi, misalnya di sini Helicopter.mp3;
8. Jalankan program *simulation_moving_virtual3Dsound*. Dalam contoh di sini disimulasikan bunyi 3D pada bidang horisontal (sudut elevasi 0°) dengan 5 sudut azimuth sebesar 270°, 315°, 0°, 45°, dan 90°. Hasil simulasi adalah file audio 3D dengan nama ‘moving3Dsound.wav’.
9. Buka *file* *moving3Dsound.wav* dengan menggunakan aplikasi audio player pada komputer, maka akan terdengar bunyi helikopter stereo yang bergerak pada bidang horisontal sesuai dengan simulasi.

Referensi

Qu, T., Xiao, Z., Gong, M., Huang, Y., Li, X., Wu, X., (2009), Distance-dependent head-related transfer functions measured with high spatial resolution using a spark gap, IEEE Trans. on Audio, Speech, and Language Processing, vol. 17, no. 6, pp. 1124-1132.