SISTEM PEMROSESAN SUARA : STUDI KASUS PEMBANDINGAN POWER SPEKTRUM DAN BISPEKTRUM PADA IDENTIFIKASI PEMBICARA MENGGUNAKAN HMM

Agus Buono [1] Benyamin Kusumoputro [2] Wisnu Jatmiko [3]

[1] Departemen Ilmu Komputer FMIPA IPB
Kampus IPB Darmaga-Bogor
email : pudesha@yahoo.co.id
[2] Fakultas Teknik Universitas Indonesia
Fakultas Teknik Kampus UI Depok
email : nynykusumo@yahoo.com
[3] Fakultas Ilmu Komputer Universitas Indonesia
Fakultas Ilmu Komputer Kampus UI Depok

ABSTRACT

Paper ini menyajikan bahasan mengenai pemrosesan sinyal suara yang meliputi paradigma pendekatan permasalahan ,lingkup kajian, perkembangan metode, hingga tahapan proses. Pada bagian akhir disajikan hasil percobaan dengan mengambil kasus identifikasi pembicara dengan teks tertentu menggunakan HMM sebagai pengenal pola. Sebagai ekstraksi ciri digunakan model MFCC dengan komponen input ada dua yang dibandingkan, yaitu power spektrum dan bispektrum.

Percobaan menggunakan data dari 10 pembicara yang mengucapkan ujaran "Pudesha" sebanyak 80 kali tanpa pengkondisian, dan disampling dengan frekuensi 11 kHz. Hasil percobaan menunjukkan bahwa pada situasi tanpa penambahan noise, teknik power spektrum mampu melakukan pengenalan dengan baik (99%). Namun dengan penambahan noise sistem gagal, bahkan dengan teknik noise canceling hasil masih rendah (74.5%) untuk noise 20 dB. Teknik bispektrum menghasilkan sistem yang lebih robust. Pada semua penambahan noise, bispektrum memberikan hasil yang lebih tinggi dibanding power spektrum. Namun dimensi bispektrum besar, sehingga proses ekstraksi ciri memerlukan waktu yang lebih lama dibanding power spektrum.

Key words

Higher Order Statistic(HOS), Mel-Frekuensi Cepstrum Coefficients (MFCC), Hidden Markov Model (HMM), Sistem Identifikasi Pembicara (SIP)

1. Pendahuluan

Seiring dengan perkembangan teknologi informasi, maka tuntutan manusia untuk memanfaatkannya guna mempermudah kehidupan sehari-hari juga makin bervariasi. Salah satu hal yang sudah dipikirkan sejak lama adalah keinginan untuk membuat komputer mampu berkomunikasi secara alami dengan manusia. Satu sistem cerdas yang pertama kali dikembangkan adalah ELIZA pada tahun 1966, yaitu suatu *artificial agent* yang mampu bercakap-cakap secara terbatas dengan user [1]. Ilmu yang membahas bidang ini dikenal dengan nama Pemrosesan Suara dan Bahasa Alami (*Speech and Natural Language Processing*). Pada pemrosesan suara lebih ditekankan pada proses ekstraksi dan pengenalannya yang bersifat bebas dari bahasa yang digunakan. Sedangkan untuk pemrosesan bahasa lebih ditekankan pada pemodelan yang terkait dengan bahasa yang digunakan.

Dari aspek metode pendekatan permasalahan, secara umum ada dua paradigma dalam bidang tersebut, yaitu paradigma stokastik dan paradigma logika formal [1]. Paradigma logika formal difokuskan pada pengembangan grammar (*metamorphosis grammar*, *definite clause grammars*, *functional grammars*) juga struktur unifikasi. Sedangkan Paradigma stokastik pada umumnya pada pemrosesan data suara, baik pada tahap praproses maupun pada pengenalan pola. Pada perkembangannya sekarang ini, bidang tersebut sudah meluas, sehingga tidak hanya masalah pemrosesan (*processing*), tetapi juga mencakup *speech understanding and generation*. Hal ini memunculkan bidang baru yang dikenal dengan Komputasi Linguistik dan Kecerdasan Buatan (*Computational Linguistics and Artificial Intelligence*), [2]. Oleh karena itu aplikasi dari bidang *Speech Processing and Understanding* menjadi luas mulai dari *document summarization* (*coding*), *transmission*, *text parsing* (*analysis*), *spelling/grammar correction* (*enhancement*), *natural language generation* (*synthesis*), *natural language*

understanding (*understanding*), *web search* (*retrieval/mining*), serta banyak lagi pada aspek *recognition* seperti mesin pendikte, identifikasi maupun verifikasi pembicara, mesin penjawab otomatis, serta interaksi menusia dengan komputer lainnya melalui suara. Selain dari aspek terapan yang begitu luas, investasi yang diperlukan lebih pada aspek *software* (bukan *hardware*), sehingga biaya yang diperlukan untuk pengembangan produk lebih murah. Hal-hal tersebut yang menjadi alasan mengapa penelitian di bidang *speech and language processing* menarik untuk dilakukan.

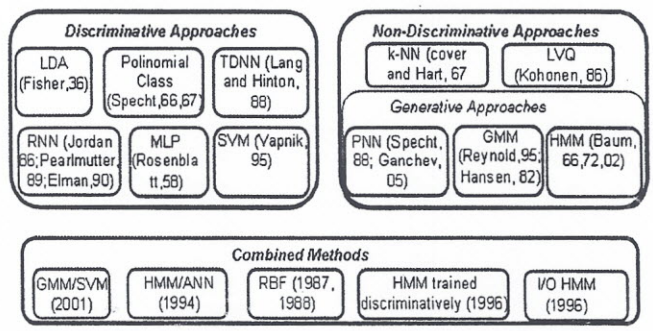
Suara merupakan satu fenomena sebagai perpaduan multidimensi, mulai dari dimensi linguistik, semantik, artikularis dan akustik [4]. Dimensi linguistik dan semantik bersifat linguistic dependent, sedangkan artikularis dan akustik bersifat linguistic independent. Dimensi artikularis secara detail menjadi kajian dari bidang fonologi, yang mempelajari bagaimana suara dan jenis-jenisnya dihasilkan. Sedangkan akustik terbagi menjadi dua bagian, yaitu yang mengkaji sinyal suara secara fisik dan bagian lainnya yang melakukan analisis terhadap sinyal suara. Oleh karena suara merupakan fungsi yang kompleks dari beberapa aspek, seperti karakteristik pembicara (dimensi titik artikularis, emosi, kesehatan, umur, jenis kelamin, dialek), bahasa, dan lingkungan (background dan media transmisi), maka pemodelan sinyal bukanlah hal yang mudah dan masih menantang untuk dilakukan.

Sebagai ilustrasi, pada paper ini akan disajikan bahasan dalam bidang pengenalan suara yang difokuskan pada sistem identifikasi pembicara. Sistem identifikasi pembicara mengenali pembicara berdasarkan suara, yang merupakan ciri biometrik seseorang yang bersifat lebih dinamis dibanding ciri biometrik lainnya, misalkan sidik jari dan tanda retina. Sifat dinamis ini disebabkan oleh beberapa hal, seperti umur, kesehatan, emosi, cara pengucapan akan menyebabkan adanya *intraspeaker variability* (variasi pada seorang pembicara). Selain masalah *intraspeaker variability*, juga adanya *noise* yang disebabkan oleh lingkungan, dan distori karena alat akan menjadi sumber error, yang pada akhirnya menurunkan akurasi sistem. Oleh karena itu, meskipun beberapa hasil penelitian telah menunjukkan akurasi yang tinggi (>95%), hal ini masih terbatas pada sinyal suara yang dikondisikan, sehingga akurasi sistem akan menurun secara nyata saat diujicobakan dalam *real life situation* [4]. Dalam kondisi real, adanya noise dan variasi internal pembicara adalah fakta yang tidak bisa dihindari, sehingga penelitian di bidang ini masih diperlukan dan layak untuk dilakukan guna memperoleh hasil yang lebih baik.

Seperti disebutkan dalam [5] bahwa persyaratan ciri biometrik sebagai pengenal seseorang, adalah bersifat alami, mudah diukur, tidak terlalu berubah dari waktu ke waktu, tidak mudah ditiru, tidak dipengaruhi kondisi phisik, serta tidak terlalu terganggu dengan adanya gangguan lingkungan. Suara adalah besaran yang hampir memenuhi semua kriteria tersebut, kecuali dua sifat terakhir, yaitu persyaratan tidak dipengaruhi kondisi phisik, serta tidak terlalu terganggu dengan adanya gangguan lingkungan. Oleh karena itu, perlu dilakukan penelitian lanjut untuk mendapatkan teknik yang mampu mengatasi masalah gangguan dikarenakan noise pada sinyal suara .

Satu permasalahan pada pengenalan suara dan hal ini juga umum terjadi pada bidang terapan lainnya adalah pada tahap ekstraksi ciri dari data masukan menjadi vektor ciri. Jika proses ekstraksi ciri dapat menghasilkan vektor ciri yang efektif mampu mencirikan obyek masukan tanpa terpengaruhi oleh adanya gangguan, maka proses pengenalan menjadi jauh lebih mudah. Telah dikenal berbagai macam teknik ekstraksi ciri yang pada dasarnya adalah memproses suatu nilai tertentu dari suara menjadi vektor ciri untuk selanjutnya sebagai input dari proses pengenalan. Besaran suara yang proses dengan teknik ekstraksi ciri tersebut merupakan barisan nilai yang didasarkan pada autokorelasi sinyal suara. Hampir semua penelitian yang ada berbasiskan pada autokorelasi orde satu yng disebut dengan power spektrum. Sejak tahun 2000, telah ada beberapa penelitian menggunakan nilai suara berbasis autuorelasi orde yang lebih tinggi dan dikenal dengan statistik orde tinggi, yaitu bispektrum (orde 2) dan trispektrum (orde 3). Hasil penelitian meunjukkan bahwa teknik berbasis statistik orde tinggi bersifat lebih robust terhadap noise dibanding dengan teknik yang berbasis power spektrum, [6].

Sedangkan pada tahapan pengenalan pola, dikenal beberapa teknik yang secara umum dikelompokkan menjadi tiga bagian, yaitu discriminative approach, non-discriminative approach dan gabungan keduanya seperti ditunjukkan pada Gambar 1, [Gan05].



Gambar 1. Pengelompokkan Teknik Pengenal Pola

Untuk memberikan gambaran proses detail dari sistem identifikasi pembicara, maka pada paper ini akan digunakan teknik ektrasi ciri menggunakan Mel-Frquency

Cepstrum Coefficients (MFCC) dan Hidden Markov Model (HMM) sebagai pengenal pola.

Selanjutnya, paper ini disajikan dengan susunan sebagai berikut : Bagian 2 mengenai pemrosesan sinyal, yang meliputi state of the art, ranah kajian, dan tahapan pemrosesan sinyal. Bagian 3 membahas pembandingan sistem yang berbasis power spektrum dengan bispektrum. Pembahasan difokuskan pada sistem identifikasi pembicara dengan HMM sebagai pengenal pola, dengan melibatkan 10 pembicara.

2. Pemrosesan Suara

2.1 State of the Art Pemrosesan Suara dan bahasa

Kajian mengenai pemrosesan suara dan bahasa (Speech and Language Processing) dilakukan diberbagai bidang, seperti Ilmu Komputer (Pemrosesan Bahasa Alami, Natural Language Processing, NLP), Ilmu Bahasa (Komputasi Linguistik, Computational Linguistics), Elektro (Speech Recognition), Psikologi (Komputasi Psikolinguistik, Computational Psycholinguistics). Sejarah perkembangan pemrosesan suara dan bahasa sejalan dengan perkembangan teknologi komputer itu sendiri, dan dibagi dalam beberapa tahap.

Tahap 1940 – 1960 : Pada tahap ini ada dua paradigma, yaitu automata (yang melandasi teori bahasa formal, formal language theory) dan probabilistik (yang melandasi model teori informasi) untuk pemrosesan suara. Model komputasi Turing mendasari munculnya automaton dan berkembang ke finite state automata dan ekspresi regular (Kleene, 1951 dan 1956). Shannon (1948) mengembangkan model probabilistik (Proses Markov Diskret) untuk pemrosesan bahasa. Hal ini diikuti oleh Chomsky (1956) yang mengembangkan finite state grammar (context-free grammar) untuk bahasa alami.Paradigma kedua adalah model komputasi probabilistik untuk pemrosesan suara dan bahasa. Pada tahap ini dikembangkan suatu metaphor untuk noisy channel dan decoding untuk transmisi bahasa melalui media komunikasi oleh Shannon. Shannon juga memperkenalkan konsep entropi dari teori termodinamika sebagai ukuran kapasitas informasi suatu channel, kandungan informasi suatu bahasa, dan pertama kali dikembangkan ukuran entropi untuk model probabilistik bahasa Inggeris. Pada tahap ini pertama kali dikembangkan spektograph (Koenig et. Al., 1946) yang memunculkan penelitian dalam bidang fonetic yang merupakan dasar dari speech recognition, dan dari sinilah ditemukan mesin speech recognizers yang pertama (1950). Pada tahun 1952 peneliti dari Bell Labs (Davis et al.) mengembangkan speaker-dependent recognizer dengan model statistik yang mampu mengenali 10 digit yang merepresentasikan dua formant pertama untuk vokal. Mesin yang dilatih dengan 10 speaker ini mampu mengenali 10 digit dari sembarang speaker dengan akurasi 97-99% yang berbasis template berdasar korelasi antara pattern dengan input.

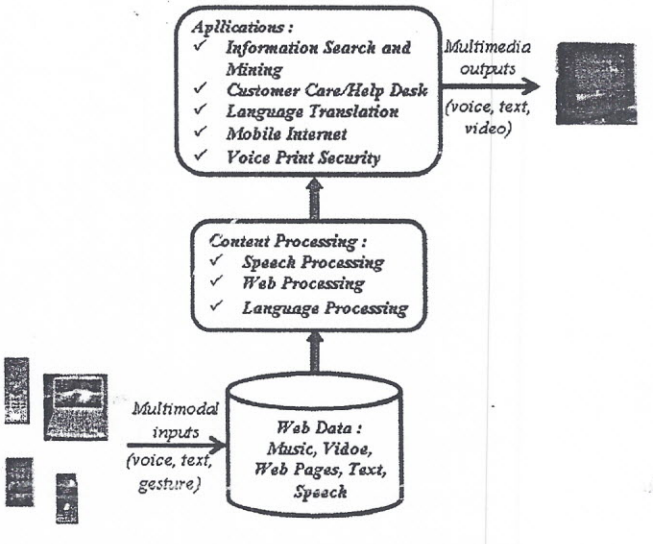
Tahap 1957 – 1970 : Penelitian bidang speech recognition berada pada dua paradigma, yaitu symbolic dan stochastic. Penelitian pada bidang symbolic banyak dilakukan oleh ahli di bidang komputer ataupun linguistik. Sedangkan penelitian pada bidang stochastic banyak dilakukan di departemen statistika ataupun electrical engineering. Jalur simbolik mengikuti penelitian dari Chomsky yang mengembangkan berbagai penelitian seperti : algoritme parsing, juga berbagai algoritme pada artificial intelligent (Joh McCarthy, Marvin Minsky, Claude shannon, dan Nathaniel Rochester). Pada tahap ini mulai dikembangkan natural language understanding sederhana yang mampu melakukan reasoning untuk memberikan jawaban pertanyaan. Pada pendekatan stochastic, mulai dikembangkan berbagai sistem seperti : optical character recognition dan text-recognition (Bledsoe dan Browning), yang menerapkan metode Bayes. Pada tahap ini mulai dikembangkan korpus yang memuat satu juta kata yang diambil dari 500 teks dari berbagai sumber (surat kabar, novel, nonfiksi, akademik, dsb.).

Tahap 1970 – 1983 : Pada tahap ini banyak sekali dilakukan penelitian mengenai speech and language processing, baik dengan paradigma stochastic maupun paradigma logic. Pada paradigma stochastic, terdapat beberapa penelitian seperti penerapan Hidden Markov Model (HMM) untuk algoritme pengenalan suara, juga metaphor dari noisy channel dan decoding, yang dilakukan secara terpisah oleh Jelinek, Bahl, Mercer, para ahli dari IBM dan dari Carneige Mellon University (CMU). Rabiner dan Juang (AT&T's Bell Labs) melakukan penelitian dalam bidang speech recognition dan synthesis. Pada paradigma logika (logic based), penelitian terutama pada pengembangan grammar (metamorphosis grammar, definite clause grammars, functional grammars) juga struktur unifikasi.Pada tahap ini dikembangkan natural language understanding yang berupa robot yang dapat menerima perintah berupa teks berbahasa alami. Penelitian pada bidang natural language understanding ini menerapkan model-model grammar, parsing, semantik dan model discourse.

Tahap 1983 – 1993 : Pada tahap ini ada dua trend yang berkembang. Pertama adalah penelitian finite-state model, seperti finite state untuk fonologi dan morfologi, juga finite state untuk sintaks. Kedua adalah penelitian empiris mengenai speech recognition

menggunakan model-model probabilistik. Penelitian yang banyak dipelopori oleh ahli dari IBM dengan pendekatan data driven ini menfokuskan pada part-of-tagging, parsing dan attachment ambiguities, dan conectionist speech recognition hingga analisis semantik.

Tahap 1994 – 1999 : Pada akhir milenium ke 20, model-model probabilistik dan data driven menjadi standar pengembangan sistem pemrosesan bahasa alami. Algoritme-algoritme untuk parsing, part-of-speech tagging, reference resolution dan pemrosesan wacana (discourse processing) didasarkan pada konsep peluang dan diterapkan pada speech recognition maupun information retrieval. Dengan berkembangnya kecepatan dan memory komputer mendorong munculnya berbagai produk mengenai speech and language processing, terutama speech recognition, spelling dan grammar checking. Pada tahap ini juga mulai muncul pemikiran mengenai information retrieval dan extraction melalui Web yang didasarkan pada bahasa (language-based information retrieval and information extraction). Pada dekade akhir abad 20 dan awal abad 21 sekarang ini, trend komunikasi pada berbagai bidang mengarah pada era digital melalui internet, maka teknologi pemrosesan suara dan bahasa menjadi mesin pendorong terjadinya perubahan cara seseorang berkomunikasi dan mengakses informasi, baik yang berupa teks, suara, video, grafik, maupun audio. Gambar 2. menyajikan protokol sederhana dari aplikasi berbasis web yang menggunakan berbagai alat akses, [GF08].

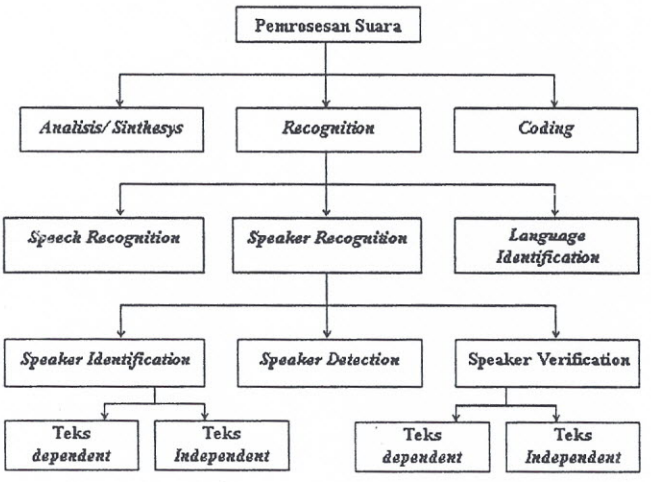


Gambar 2. Contoh Protokol Aplikasi Berbasis Web

Beberapa aplikasi yang menantang, seperti *voice search* yang mengabungkan *automatic speech recognition* dengan *document search*, *spoken document retrieval* (SDR), dan *spoken language understanding and translation (bilingual* dan *multilingual)* belum menemukan solusi yang optimun, sehingga bidang ini masih terbuka lebar untuk pengembangan riset

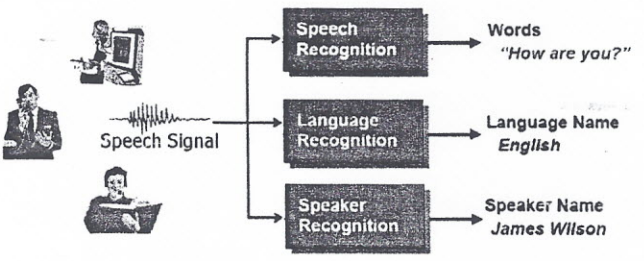
2.2 Ranah Kajian Pemrosesan Suara

Camphell, 1997, menyebutkan bahwa pemrosesan suara mempunyai tiga ranah kajian, yaitu masalah sinthesis, pengenalan dan coding, seperti yang sajikan pada Gambar 3., [3].



Gambar 3. Cakupan Kajian dalam Pemrosesan Suara

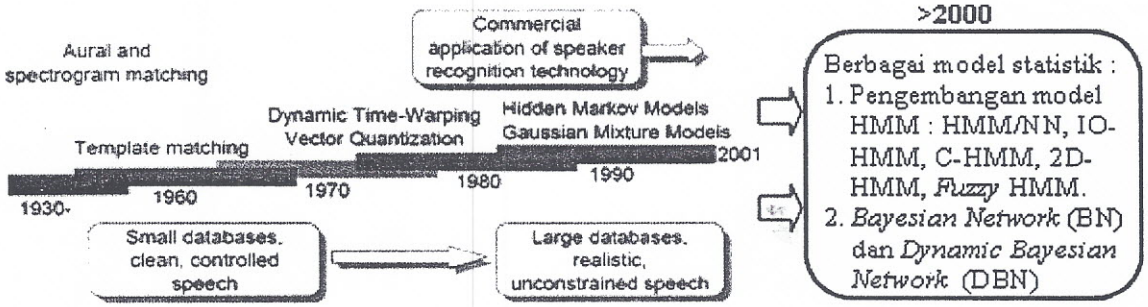
Kajian bidang *Recognition* diarahkan pada pemrosesan suara untuk klasifikasi yang secara umum terdiri dari dua sub sistem, yaitu ekstraksi ciri dan pengenal pola. Sesuai keluaran sistem, bidang *recognition* dapat dipisahkan menjadi tiga ranah kajian, yaitu pengenalan suara (*speech recognition*), pengenalan pembicara (*speaker recognition*), dan identifikasi bahasa (*language identification*). Perbedaan dari ketiga area tersebut diperlihatkan pada Gambar 4., [5].



Gambar 4. Kajian pada Ranah *Recognition*

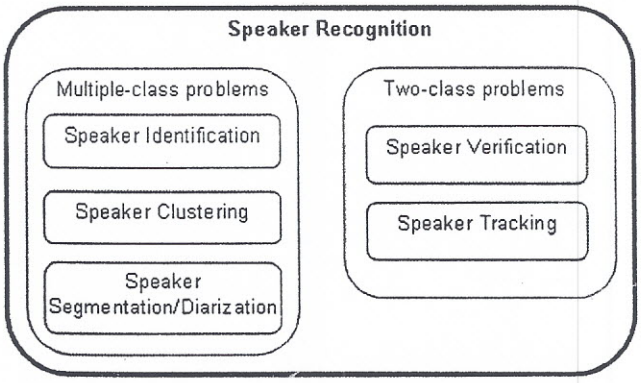
Hal yang dilakukan pada area pengenalan (*recognition*) adalah untuk mengekstrak informasi yang

terkandung di dalam sinyal suara, [5]. Penelitian di bidang pengenalan ini telah terjadi sejak tahun 1960 mulai dari pemodelan yang bersifat determistik hingga probabilistik seperti disajikan pada Gambar 5., (dimodifikasi dari [5]).



Gambar 5. Perkembangan Penelitian Bidang Pengenalan Suara

Pengenalan Pembicara (*Speaker Recognition*) merupakan suatu proses yang secara otomatis mengenali siapa pembicara (*who is speaking*) menggunakan informasi spesifik yang terkandung pada sinyal suara, [7]. Berdasar keluaran sistem, Gancev membagi sistem pengenalan pembicara menjadi dua, yaitu *multiple-class problem* dan *two-class problem* [8], seperti disajikan pada Gambar 6.



Gambar 6. Sistem Pengenalan Pembicara Berdasar Jumlah Kelas Output

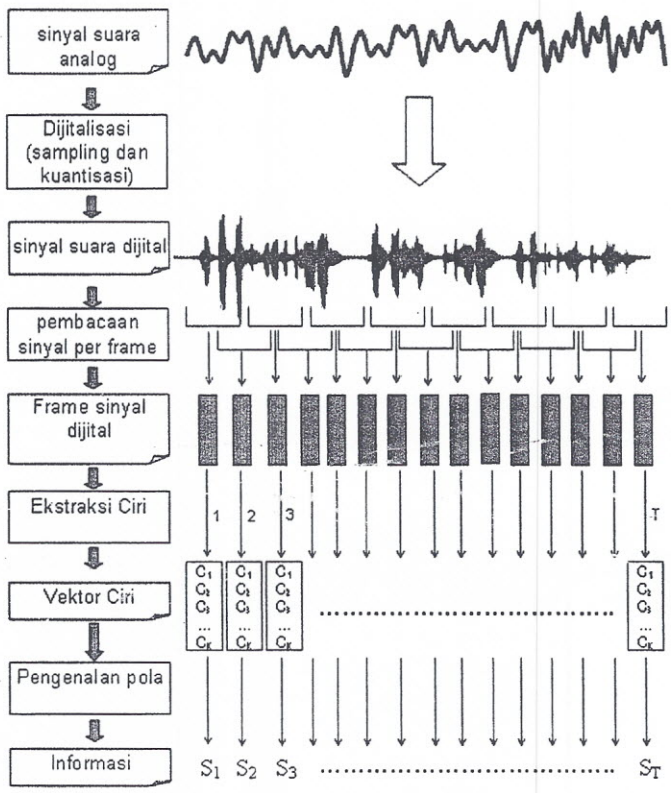
Speaker Verification adalah suatu permasalahan dua kelas, yaitu sistem akan menolak atau menerima suatu klaim mengenai identitas seorang pembicara dengan berdasarkan data suara yang diberikan. Sementara itu, pada speaker tracking /detection, sistem akan mendeteksi bagian atau segmen waktu yang merupakan suara seorang pembicara tertentu dari sebuah segmen sinyal suara dengan durasi tertentu. *Speaker segmentation* akan memberikan label pada setiap segmen tertentu dari sinyal input sesuai kode pembicara yang paling sesuai. *Speaker clustering* melakukan pengelompokkan sehingga sinyal suara yang mirip ada dalam satu kelompok yang merupakan milik seorang pembicara atau kelas pembicara ke dalam database. Sedangkan *speaker identification* adalah menentukan pembicara yang paling mungkin dari sinyal suara yang diberikan. Kajian yang akan dilakukan pada penelitian ini adalah pada bidang *speaker recognition* dan difokuskan pada *speaker identification* atau identifikasi pembicara.

2.3 Transformasi Sinyal Menjadi Informasi

Dalam [9] disebutkan bahwa sinyal suara merupakan gelombang longitudinal yang tercipta dari tekanan udara yang berasal dari paru-paru yang berjalan melewati lintasan suara menuju mulut dan rongga hidung dengan bentuk artikulator yang senantiasa berubah. Pemrosesan suara merupakan teknik mentransformasi gelombang longitudinal tersebut menjadi informasi yang berarti sesuai yang diinginkan. Secara umum proses transformasi tersebut terdiri dari dijitalisasi sinyal analog, ekstraksi ciri dan diakhiri dengan pengenalan pola untuk klasifikasi, seperti diilustrasikan pada Gambar 7.

Sesuai dengan Gambar 7. tersebut, maka proses transformasi sinyal suara menjadi informasi yang akan dijelaskan pada Sub Bab ini disajikan dalam tiga konsep, yaitu mengenai sinyal, ekstraksi ciri, dan pengenalan pola. Pada bagian sinyal, pembahasan dimulai dengan terminologi sinyal, ukuran kualitas sinyal, sampling dan kuantisasi, serta pembacaan sinyal untuk pemrosesan. Untuk ekstraksi ciri, pembahasan difokuskan pada teknik *mel-frequency cepstrum coefficients* (MFCC). Hal ini dengan pertimbangan teknik tersebut relatif lebih baik dibanding teknik lain yang sudah ada. Untuk pengenalan pola, akan difokuskan pada model *hidden markov model*, (HMM). Hal ini didasarkan fakta bahwa HMM merupakan model yang menjadi *trend* serta paling banyak dikaji pada riset terbaru mengenai pemrosesan sinyal. Algoritme detail mengenai HMM (algoritme *forward*, *backward*, *viterbi*, *k-means* serta algoritme *Baum-Welch*) disajikan secara terpisah pada lampiran disertasi ini. Algoritme *forward* dan *backward* dipergunakan untuk menghitung peluang barisan observasi, algoritme *viterbi*

untuk menduga barisan *hidden state* yang optimum dan algoritme *k-means* dan *Baum-Welch* untuk menduga parameter HMM.



Gambar 7. Alur Proses Transformasi Sinyal Suara Analog Menjadi Informasi

Digitalisasi Sinyal Suara

Digitalisasi sinyal suara bertujuan untuk mengubah sinyal analog menjadi sinyal digital. Pada digitalisasi sinyal suara ini ada tiga proses, yaitu sampling, kuantisasi dan coding. Sampling merupakan pengamatan nilai sinyal waktu kontinyu (sinyal analog) pada suatu waktu tertentu, sehingga diperoleh sinyal waktu diskret. Banyak cara untuk melakukan sampling pada sinyal analog. Salah satu yang sering digunakan adalah *periodic* atau *uniform sampling*. Dalam hal ini sampling dilakukan pada setiap selang waktu yang tetap, yaitu pada setiap selang waktu T. Hubungan antara sinyal waktu diskret hasil sampling dengan sinyal analog adalah sebagai berikut :

$x(n)=x_a(nT)$ dengan $-\infty<n<\infty$

x(n) adalah sinyal waktu diskret yang diperoleh dari sampling terhadap sinyal waktu nyata $x_a(t)$ setiap T detik. Dalam hal ini T sebagai periode sampling, dan $F_s=1/T$ adalah sampling rate (Hertz). Sebagai ilustrasi, misalkan sinyal analog disampling dengan sampling rate 11 kHz, ini berarti setiap detik disampling sebanyak 11000 kali. Dengan kata lain, setiap detik dicatat nilai simpangan sinyal sebanyak 11000 data. Atau dengan kata lain lagi, jarak antara satu data simpangan dengan data berikutnya adalah 1/11000 detik. Permasalahan dalam sampling adalah bagaimana menentukan T atau juga F_s dari suatu sinyal analog dengan frekuensi F.

Sinyal sinusoid analog dengan frekuensi F untuk setiap waktu t dirumuskan sebagai :

$$x_a(t) = A\cos(2\pi Ft + \theta)$$

Oleh karena itu :

$$x_a(nT) = A\cos(2\pi FnT + \theta) = A\cos\left(\frac{2\pi nF}{F_s} + \theta\right)$$

Sementara itu sinyal diskret sinusoid, x(n), dapat dirumuskan sebagai :

$$x(n) = A\cos(2\pi fn + \theta)$$

Dari tiga persamaan terakhir, terlihat bahwa $f = \frac{F}{F_s}$, yang dapat diartikan sebagai jumlah gelombang per sample. Misalkan F=10Hz (yang berarti 10 gelombang per detik) dan F_s=5 sample per detik, maka f=10/5=2 gelombang per sample. Berdasar persamaan inilah, nilai F_s akan ditentukan sehingga semua komponen frekuensi dalam sinyal dapat direpresentasikan secara khas. Sesuai sifat kosinus, maka persamaan ketiga di atas juga bisa ditulis sebagai :

$$x(n) = A\cos(2\pi fn + \theta) = A\cos(2\pi fn + \theta)$$
$$= A\cos(2\pi fn + 2k\pi n + \theta)$$

untuk k=0,1,2,3,.... Hal ini berarti bahwa sinyal diskret dengan kecepatan sudut $\omega_k = 2\pi f + 2k\pi$ tidak dapat dibedakan dengan sinyal diskret dengan kecepatan sudut $\omega_0 = 2\pi f$. Sesuai sifat kosinus, sinyal diskret yang dapat dibedakan hanya pada rentang kecepatan sudut $-\pi < \omega_0 = 2\pi f < \pi$ atau dengan kata lain -1/2<f<1/2.

Oleh karena karena $f = \frac{F}{F_s}$, maka sampling dengan frekuensi sampling F_s hanya mampu memberikan hasil yang berbeda untuk sinyal-sinyal kontinyu dengan frekuensi F<0.5F_s. Sebagai gambaran, misalkan ada dua sinyal dengan frekuensi F1=10 Hz ($x1(t) = \cos 2\pi 10t$) dan F2=50 Hz ($x2(t) = \cos 2\pi 50t$), maka kalau masing-masing disampling dengan F_s=40sample per detik akan dihasilkan sinyal diskret :

$$x1(n) = \cos 2\pi \frac{10}{40} n = \cos \frac{\pi}{2} n$$

$$x2(n) = \cos 2\pi \frac{50}{40} n = \cos \frac{5\pi}{2} n$$

$$= \cos(2\pi n + \frac{\pi}{2} n) = \cos \frac{\pi}{2} n$$

Terlihat sampling dari dua sinyal analog tersebut dengan F_s=40Hz akan menghasilkan dua sinyal diskret yang sama persis. Hal ini disebut bahwa sinyal dengan frekuensi 50 Hz adalah alias dari sinyal dengan dengan frekuensi F=10 Hz dengan sampling rate 40 sample per detik.

Oleh karena sinyal analog dapat direpresentasikan sebagai penjumlahan dari gelombang sinus dengan amplitudo, frekuensi dan fase yang berbeda, yang dalam hal ini sebanyak N komponen sesuai formula berikut :

$$x_a(t) = \sum_{i=1}^{N} A_i(t) \sin[2\pi F_i(t) + \theta_i(t)] \quad (2.11)$$

maka nilai sampling rate yang dapat menangkap semua komponen sinyal haruslah minimal dua kali frekuensi maksimum yang ada dalam sinyal. Nilai sampling rate sebesar $F_s=2F_{max}$ disebut sebagai *Nyquist rate*.

Kuantisasi merupakan proses mengkonversi nilai amplitudo yang bersifat kontinyu pada suatu titik waktu tertentu menjadi sinyal digital dengan mengekspresikannya menggunakan sejumlah digit tertentu. Konversi nilai kontinyu menggunakan sejumlah digit ini akan menghasilkan *error* yang disebut *quantization error* atau *quantization noise*.

Secara umum, proses kuantisasi dilakukan dengan pembulatan ke nilai terdekat (*rounding*), atau bisa juga dengan pemotongan bagian sisa (*truncating*). *Error* karena kuantisasi dengan metode pembulatan, $e_q(n)$, adalah pada selang :

$$-\frac{\Delta}{2} \le e_q(n) \le \frac{\Delta}{2}, \text{ dengan } \Delta = \frac{x_{max} - x_{min}}{L-1}$$

L adalah banyaknya level kuantisasi, x_{max} dan x_{min} adalah nilai maksimum dan minimum yang akan dikuantisasi. Dalam hal ini *error* kuantisasi merupakan selisih antara nilai sinyal analog dengan nilai hasil kuantisasinya, yaitu :

$$e_q(t) = x_a(t) - x_q(t)$$

Ukuran kualitas output dari suatu mesin konversi analog ke digital (A/D *converter*) biasanya diukur dengan *signal-to-quantization noise ratio* (*SQNR*) yang dinyatakan sebagai rasio energi signal terhadap energi *noise*, yaitu **[10]** :

$$SQNR = \frac{P_x}{P_q} = \frac{3}{2} 2^{2b} \quad (2.13)$$

Dengan satuan *deciBell*, *dB*, maka SQNR dirumuskan sebagai :

$$SQNR(dB) = 10\log_{10}(SQNR) = 1.76 + 6.02b \quad (2.14)$$

Ini berarti setiap penambahan 1 bit pada representasi digital, akan meningkatkan SQNR sekitar 6 dB. Sebagai contoh pada *compact disc player*, menggunakan representasi 16 bit, sehingga nilai SQNR adalah lebih dari 96 dB.

Coding merupakan pemberian bilangan biner pada setiap level kuantisasi. Jika kuantisasi yang diterapkan mempunyai level sebanyak L, maka setidaknya harus tersedia L bilangan biner yang berbeda. Sedangkan kode biner dengan panjang b akan dapat menghasilkan kode berbeda sebanyak 2^b. Oleh karena itu untuk kuantisasi dengan L level diperlukan bilangan biner dengan panjang $b \geq \log_2 L$.

Pembacaan Sinyal digital

Untuk keperluan pemrosesan, sinyal analog yang sudah didijitalkan (dengan sampling dan kuantisasi) dibaca dari frame demi frame dengan lebar tertentu yang saling tumpang tindih. Panjang frame ini biasanya 5 hingga 100 *milisecond* dengan overlap antar frame yang berurutan adalah 0, 25, 50 atau 75%. Proses ini dikenal dengan *frame blocking*. Satu frame tersebut sebagai satu unit terkecil yang mengandung satu unit informasi, sehingga barisan frame akan menyimpan suatu informasi yang lengkap dari sebuah sinyal suara. Untuk itu, distori antar frame harus diperkecil atau diminimalisasi. Satu teknik untuk meminimalkan distorsi antar frame adalah dengan melakukan proses filtering pada setiap frame. Secara umum fungsi filtering ada dua, yaitu untuk memisahkan sinyal dari berbagai sumber lain yang "mengotori" serta untuk "menjernihkan" sinyal dari adanya distorsi. Secara umum dikenal enam jenis filter seperti disajikan pada Tabel 1.

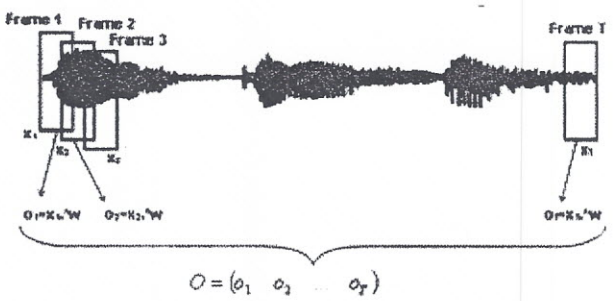
Tabel 1. Klasifikasi Filtering

Domain Penerapan	Metode Penerapan	
	Konvolusi (FIR)	Rekursif (IIR)
Domain waktu (smoothing, DC removal)	Moving Average	Single pole
Domain Frekuensi (memisahkan frekuensi)	Windowing	Chebyshev
Kustomisasi (dekonvolusi)	FIR custom	Iterative design

Gambar 8 memberikan ilustrasi proses filtering dengan fungsi window **w**. Jika sinyal dijital frame ke i adalah $\mathbf{x}_i$ dan fungsi window yang digunakan adalah **w**, maka output windowing frame ke i adalah $\mathbf{y}_i = \mathbf{x}_i.'\mathbf{w}$, yaitu perkalian setiap komponen yang seletak dari vektor $\mathbf{x}_i$ dengan vektor **w**.

Ekstraksi Ciri

Ekstraksi ciri merupakan proses untuk menentukan satu nilai atau vektor yang dapat dipergunakan sebagai penciri obyek atau individu. Di dalam pemrosesan suara,



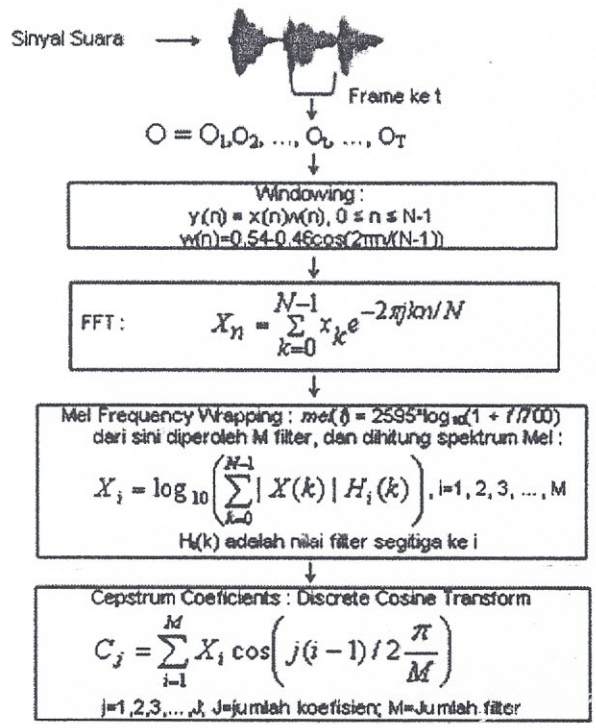
Gambar 8. Proses *Frame Blocking* dan

ciri yang biasa dipergunakan adalah nilai koefisien cepstral dari sebuah frame. Satu teknik ekstraksi ciri sinyal suara yang umum dan menunjukkan kinerja yang baik adalah teknik *Mel-Frequency Cepstrum Coefficient*,(MFCC) yang menghitung koefisien cepstral dengan mempertimbangkan persepsi sistem pendengaran manusia terhadap frekuensi suara. Dibandingkan dengan metode ekstraksi ciri lainnya, Davis dan Mermelstein memperlihatkan bahwa MFCC sebagai teknik ekstraksi ciri memberikan hasil pengenalan yang tinggi, [8]. Setelah diperkenalkannya teknik ini, berbagai variasi telah dikembangkan, terutama dalam hal jumlah, bentuk, dan lebar filter serta cara membentuk intervalnya. *Mel-Frequency Cepstrum Coefficient*, (MFCC) sebagai pengekstraksi ciri dan teknik untuk parameterisasi sinyal suara telah banyak digunakan pada berbagai bidang area pemrosesan suara, terutama pada sistem identifikasi pembicara. Diagram alur teknik MFCC dalam mengekstrak sinyal suara adalah seperti pada Gambar 9., [8].

Dari Gambar 9. terlihat bahwa sinyal dibaca frame demi frame, dan dilakukan windowing untuk setiap frame untuk berikutnya dilakukan transformasi Fourier. Dari nilai hasil transformasi Fourier ini selanjutnya dihitung spektrum mel menggunakan sejumlah filter yang dibentuk sedemikian sehingga jarak antar pusat filter adalah konstan pada ruang frekuensi mel. Dari literatur yang ada, skala mel ini dibentuk untuk mengikuti persepsi sistem pendengaran manusia yang bersifat linear untuk frekuensi rendah dan logaritmik untuk frekuensi tinggi, dengan batas pada nilai frekuensi akustik sebesar 1000 Hz. Koefisien MFCC merupakan hasil transformasi *Cosinus* dari spektrum mel tersebut, dan dipilih K koefisien. Transformasi kosinus berfungsi untuk mengembalikan domain, dari frekuensi ke domain waktu.

Pengenalan Pola

Keluaran ekstraksi ciri ini akan masuk ke sub sistem classifier untuk dilakukan proses pengenalan. Dalam hal ini ada dua tipe pengenal, yang pertama adalah seluruh hasil ekstraksi ciri dari semua frame pada frase ujaran diproses secara bersama-sama (digabungkan atau mungkin juga dirata-ratakan sehingga menjadi satu vektor ciri) menjadi masukan sub sistem classifier untuk dikenali.



Gambar 9. Diagram Alur Teknik MFCC untuk Mengekstrak Sinyal

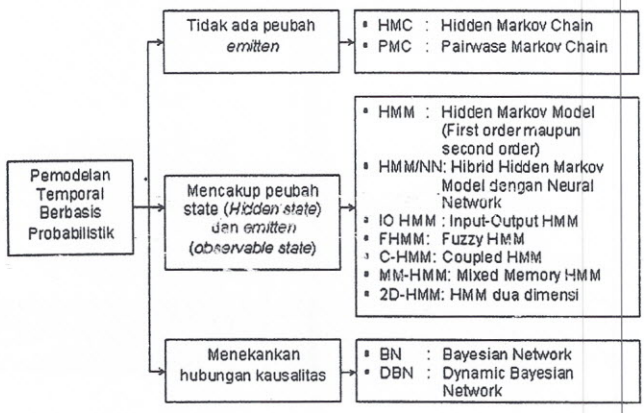
Jenis pengenal yang melakukan proses seperti ini misalnya adalah template matching dan neural network. Namun demikian, ada juga tipe pengenal yang membaca sinyal masukan frame demi frame sesuai periode diujarkannya, dan setelah semua frame diproses, baru diberikan skor bagi sinyal. Tipe kedua ini dilakukan oleh pemodelan temporal/spasial seperti misalnya yang dilakukan oleh model berbasis proses Markov.

Dalam perkembangannya, model berbasis proses Markov menjadi trend dari sistem pemrosesan sinyal, khususnya pada sub sistem *classifier*, sehingga telah dikenal berbagai macam variasi dari model Markov tersebut, seperti disajikan pada Gambar 10. Pada gambar tersebut, yang dimaksud peubah *emitten* adalah peubah yang kemunculannya sebagai efek dari peubah lain yang tidak dapat diobservasi secara langsung yang disebut peubah *state*.

Pada pemodelan yang tidak melibatkan peubah emitten, barisan peubah acak dimodelkan mengikuti proses markov, dan hal ini dikenal dengan Hidden Markov Chain, HMC. Andaikan peubah acak tersebut muncul secara berpasangan, maka fenomena tersebut bisa dimodelkan dengan model Pairwase Markov Chain, PMC. Sesuai dengan [11], PMC didefinisikan sebagai berikut : Kalau X=(X1, X2, X3, ..., Xn) dan Y=(Y1, Y2, Y3, ..., Yn) adalah dua barisan peubah acak temporal berpasangan dan ditulis sebagai Z==(Z1, Z2, Z3, ..., Zn) dengan Zi=(Xi,Yi), maka Z adalah PMC berkaitan dengan X dan Y jika distribusi Z dapat dirumuskan sebagai :

$$p(z) = \frac{p(z_1, z_2)p(z_2, z_3)...p(z_{n-1}, z_n)}{p(z_2)p(z_3)...p(z_{n-1})}$$

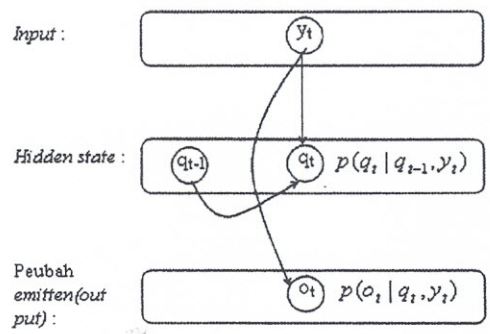
Dengan $p(z_i)$ merupakan distribusi marjinal dari $p(z_{i-1}, z_i)$ dan juga sebagai distribusi marjinal dari $p(z_i, z_{i+1})$. Pemakaian model PMC pada kasus segmentasi citra yang terdistori oleh noise memberikan hasil yang lebih baik dibanding dengan model HMC, seperti yang ditunjukkan pada **[12]**.



Gambar 10. Pemodelan Temporal Berbasis Probabilistik

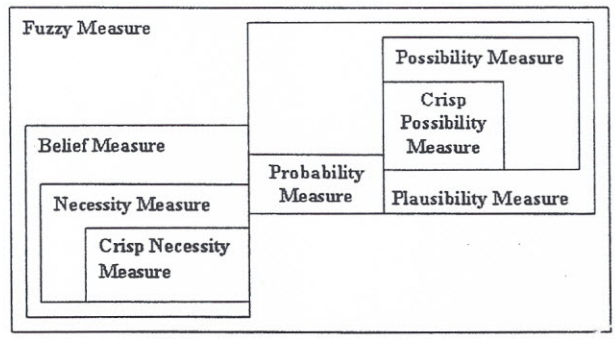
Pada suatu kondisi tertentu, nilai-nilai peubah X pada proses Markov di atas tidak teramati secara langsung (disebut sebagai unobservable variable atau unobservable state atau hidden state), namun dapat dievaluasi dari peubah lain yang dapat diamati secara langsung (disebut observable variable atau peubah emitten) yang merupakan efek dari peubah tak teramati X tersebut. Sebagai contoh misalkan pada bidang kesehatan, kondisi jantung (tidak teramati) seorang pasien dapat dievaluasi berdasar tekanan darah, suhu ataupun peubah emitten lain yang dapat diobservasi langsung. Untuk situasi seperti ini, model Markov diperluas menjadi Model Markov Tersembunyi (Hidden Markov Model, HMM). Oleh karena itu, selain parameter peluang transisi antar state, juga diperlukan distribusi untuk peubah emitten yang dalam hal ini merupakan peluang bersyarat (conditional probability). Jika distribusi peluang untuk peubah observasi (emitten) ini diduga dengan menggunakan jaringan syaraf tiruan (neural network), maka dikenal menjadi model HMM/NN. Penerapan teknik HMM/NN ini sebagai classifier pada speaker recognition dapat dilihat pada **[8]**. Sedangkan neural network secara tersendiri untuk identifikasi pembicara dapat dilihat pada **[13]**.

IO-HMM merupakan model HMM yang dalam hal ini distribusi peubah *emitten* selain sebagai *influence* dari peubah *state*, juga tergantung dari input yang diberikan. Oleh karena itu, pada IO-HMM ada tiga barisan peubah, yaitu input, *state* dan *emitten* yang juga disebut sebagai output, seperti disajikan pada Gambar 11.



Gambar 11. Hubungan *Input*, *Hidden State* dan *Emitten* pada IO-HMM

Variasi lain dari model HMM adalah seperti yang dilakukan oleh Mohamed dan Gader pada **[14]** yang menggunakan konsep ukuran kekaburan (*fuzzy measure*) yang dalam hal ini adalah λ-*measure* sebagai ukuran ketidakpastian (*uncertatinty measure*). Model yang dikembangkannya ini disebut sebagai *Fuzzy Hidden Markov Model*, FHMM. Pada HMM biasa, ukuran ketidakpastian menggunakan konsep peluang (*probability measure*), yang merupakan kasus khusus dari ukuran kekaburan secara umum, seperti diperlihatkan pada Gambar 12.



Gambar 12. Hubungan antara Berbagai Ukuran Ketidakpastian

Oleh karena itu model FHMM yang dikembangkan oleh Magdi dan Gader disebutnya sebagai Generalisasi HMM. Model FHMM ini diimplementasikan untuk melakukan pengenalan terhadap tulisan tangan dan memberikan peningkatan akurasi dari 94.3% menjadi 95.6%, **[15]**. Sementara itu Hosseyndoost dan Teshnehlab, 2005, mengembangkan model fuzzy HMM yang berbeda dengan model FHMM sebelumnya, dan dipergunakan untuk klasifikasi fonem dan *phonetic transcription* dengan persentase kesalahan berkisar dari 30 hingga 39%, **[16]**. Model fuzzy HMM yang dikembangkan pada **[16]** melakukan modifikasi pada

peluang distribusi peubah emitten, yaitu dengan memberikan semacam pembobot fuzzy.

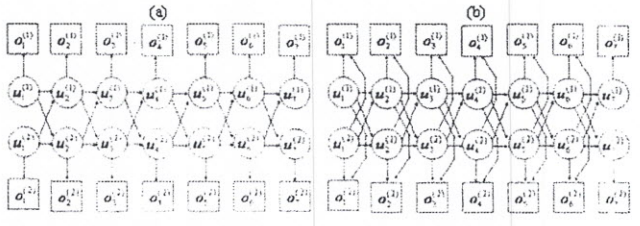
Pada situasi barisan output yang dihasilkan adalah berpasangan, maka model HMM dikembangkan menjadi *Coupled* HMM (C-HMM) atau bisa juga *Mixed Memory* HMM (MM-HMM). Dalam hal ini barisan $\mathbf{O}^{(1)}$ dan $\mathbf{O}^{(2)}$, yaitu :

$$O^{(1)} = \left(o_0^{(1)}, o_1^{(1)}, o_2^{(2)}, ..., o_{T-1}^{(1)}\right)$$
$$O^{(2)} = \left(o_0^{(2)}, o_1^{(2)}, o_2^{(2)}, ..., o_{T-1}^{(2)}\right)$$

dipandang sebagai emitten dari barisan hidden state :

$$U^{(1)} = \left(u_0^{(1)}, u_1^{(1)}, u_2^{(2)}, ..., u_{T-1}^{(1)}\right)$$
$$U^{(2)} = \left(u_0^{(2)}, u_1^{(2)}, u_2^{(2)}, ..., u_{T-1}^{(2)}\right)$$

Dua pendekatan untuk memodelkan hubungan antara emitten dan hidden state di atas adalah menggunakan C-HMM dan MM-HMM. Gambar 13. memberikan ilustrasikan latice dari model tersebut untuk sekuen dengan panjang tujuh, [17]. Untuk kasus yang sama, Pan dan Liang 2004, [17], mengembangkan model HMM yang disebut sebagai Fused HMM yang pada intinya adalah mengformulasikan peluang observasi gabungan dengan konsep jarak antar distribusi yang diukur dengan entropi relatif menggunakan *Kullback-Leibler divergence.*



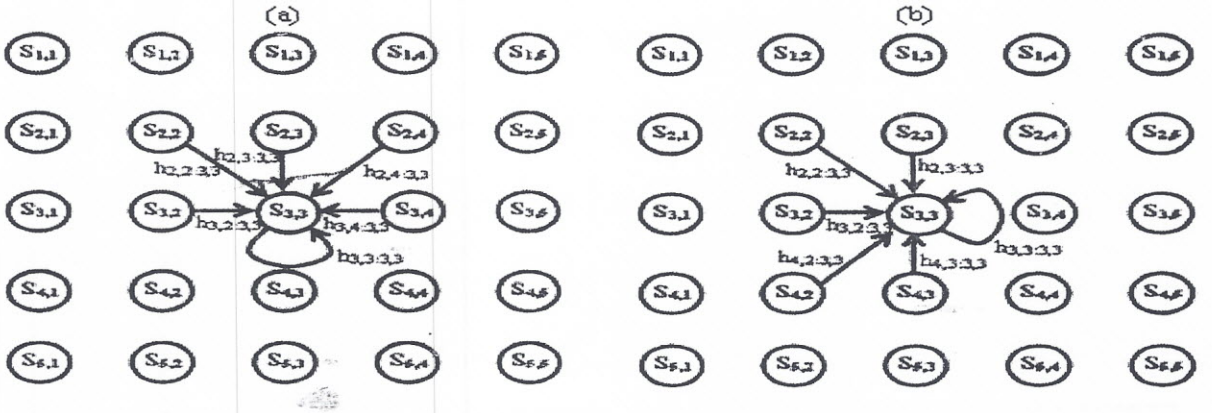
Gambar 13. Latice untuk Variasi Model HMM dengan Panjang Tujuh, (a) C-HMM, (b) MM-HMM

Model-model HMM tersebut mengacu pada kasus yang memandang bentuk sekuen pada satu arah (horisontal). Pada kasus citra, sekuen data akan lebih baik dipandang pada dua arah (horisontal dan vertikal). Untuk mengakomodasi dua cara pandang ini, maka dikembangkan 2D HMM. Gambar 14. menyajikan transisi vertikal dan horisontal 2D HMM ber state 5x5, [18].

Keseluruhan model yang sudah dibahas di atas lebih didasarkan pada dua jenis peubah, yaitu *hidden state* dan *observable state.* Selain itu dikenal model probabilistik lain yang lebih menekankan pada hubungan kausalitas antar peubah yang dikenal dengan *Bayesian Network*, BN. BN merupakan suatu *Directed Acyclic Graph* dengan setiap node mewakili peubah dalam sistem, serta *link*(A,B) menyatakan adanya pengaruh langsung dari peubah A terhadap peubah B, serta pada setiap node B mempunyai distribusi peluang bersyarat, P(B|parent(B)). Jika model BN ini dikembangkan mengikuti indeks waktu, maka dikenal dengan nama *Dynamic Bayesian Netwrok*, DBN

Teknik HMM Sebagai Pengenal Pola

Hidden Markov Model (HMM) merupakan model markov orde satu yang mempunyai dua jenis state, yaitu hidden state dan observable state. Setiap *hidden state* dapat menghasilkan suatu *outcome* yang teramati pada setiap periode t, yaitu O_t. *Outcome* dari *hidden state* ini disebut sebagai *observable state* atau *emitten*. Oleh karena itu, dari periode t=1 hingga t=T diperoleh barisan peubah teramati (*observation state*) $O=O_1, O_2, O_3, \ldots, O_T$, yang merupakan *outcome* dari barisan peubah tak teramati $Q=q_1, q_2, q_3, \ldots, q_T$. Berdasar hubungan antar *state*, dikenal dua jenis HMM, yaitu *ergodic* dan *left-right* HMM. Pada *Ergodic* HMM, antar dua state selalu ada *link*, sehingga disebut juga sebagai *fully connected* HMM. Sedangkan pada *left-right* HMM, state dapat disusun dari kiri ke kanan sesuai dengan *link*-nya. Gambar 15. memberikan contoh *ergodic* dan *left-right* HMM dengan tiga *hidden state* dengan distribusi peubah *emitten*-nya



Gambar 14. Contoh Transisi Vertikal (a) dan Horisontal (b) ke State $S_{3,3}$ ($h_{i,j:x,y}$ adalah peluang transisi dari state $S_{i,j}$ ke state $S_{x,y}$)

adalah *Gaussian.*

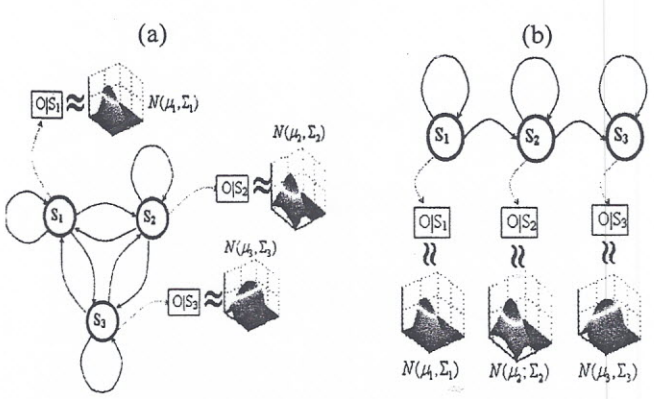
Untuk mempermudah perumusan matematika, berikut disajikan notasi-notasi mengenai HMM [19].

T : Panjang barisan observasi atau panjang periode pengamatan.
N : Banyaknya kemungkinan nilai *hidden state*.
S : Himpunan nilai-nilai *state* yang mungkin, $S=\{S_1, S_2, S_3, \ldots, S_N\}$
Q : $(q_1, q_2, q_3, \ldots, q_T)$ adalah barisan *state* dari periode ke 1 hingga T, q_t adalah *state* yang dikunjungi pada periode t
M : Banyaknya kemungkinan kemunculan peubah teramati
V : Himpunan kemungkinan observasi, $V=\{ v_1, v_2, v_3, \ldots, v_M\}$
Л : Adalah himpunan $\{л_i\}$, dengan $л_i=P(q_1=i)$, yaitu peluang pada tahap awal berada pada *state* i. Dalam hal ini berlaku $\sum_{i=1}^{N} \pi_i = 1$
A : Adalah himpunan $\{a_{ij}\}$ dengan $a_{ij}=P(q_{t+1}=S_j|q_t=S_i)$, yaitu peluang berada di *state* S_j pada waktu t+1, kalau pada waktu t berada di *state* S_i. Dalam hal ini diasumsikan a_{ij} bebas dari waktu.
B : Adalah himpunan $\{b_j(k)\}$, dengan $b_j(k)=P(v_k$ pada waktu $t|q_t=S_j)$, yaitu peluang peubah teramati yang muncul adalah simbol v_k kalau *state* yang terjadi adalah S_j.
O : $(O_1, O_2, O_3, \ldots, O_T)$ adalah barisan observasi, dengan O_t sebagai nilai atau vektor yang teramati (*observable symbol*) pada waktu t.

Suatu HMM dinotasikan dengan :

$$\lambda = (A, B, \Pi)$$

A adalah matriks peluang transisi, B adalah matriks peluang observasi dan Л adalah vektor peluang awal.



Gambar 15. Contoh HMM dengan Tiga Hidden State dan Distribusi Emitten *Gaussian*, (a) *Ergodic*, (b) *Left-Right* HMM

Dari sebuah model HMM, dikenal tiga algoritme sesuai problem yang akan dijawab, yaitu algoritme evaluasi yang dipergunakan untuk menduga peluang kemunculan sebuah barisan observasi, algoritme pelatihan untuk menduga nilai-nilai parameter HMM dan algoritme *decode* untuk menduga kemungkinan barisan *state*. Berikut disajikan perumusan ke tiga algoritme tersebut.

1. **Problem 1 (Evaluation) :** Untuk suatu $\lambda = (A,B,Л)$ tertentu, ingin diketahui $P(O|\lambda)$, yaitu peluang munculnya barisan $O= O_1, O_2, O_3, \ldots, O_T$.
   Solusi :
   Barisan $O = O_1, O_2, O_3, \ldots, O_T$ adalah nilai teramati yang merupakan refleksi atau *emitten* dari barisan *hidden state* $Q = q_1, q_2, \ldots, q_T$. Untuk suatu barisan *hidden state* tertentu, $Q = q_1, q_2, \ldots, q_T$, nilai $P(O|\lambda)$ dapat dihitung dengan penurunan berikut :
   $$P(O \mid Q, \lambda) = \prod_{t=1}^{T} P(O_t \mid q_t, \lambda)$$
   Dengan asumsi kebebasan setiap antar observasi, maka nilai tersebut dapat dirumuskan menjadi :
   $$P(O \mid Q, \lambda) = b_{q_1}(O_1) \cdot b_{q_2}(O_2) \cdots b_{q_T}(O_T)$$
   Sedangkan peluang kemunculan barisan $Q = q_1, q_2, \ldots, q_T$ tertentu adalah :
   $$P(Q \mid \lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} a_{q_3 q_4} \cdots a_{q_{T-1} q_T}$$
   Distribusi bersama O dengan Q diperoleh dengan mengalikan keduanya :
   $$P(O, Q \mid \lambda) = P(O \mid Q, \lambda) \cdot P(Q \mid \lambda)$$
   Oleh karena itu nilai $P(O|\lambda)$ diperoleh dengan menjumlahkan formula di atas untuk semua kombinasi barisan *hidden state* yang mungkin.
   $$P(O \mid \lambda) = \sum_{\forall Q} P(O \mid Q, \lambda) P(Q \mid \lambda)$$
   $$= \sum_{\forall Q} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \ldots a_{q_{T-1} q_T} b_{q_T}(O_T)$$
   $$= \sum_{\forall Q} \pi_{q_1} b_{q_1}(O_1) \prod_{t=2}^{T} a_{q_{t-1} q_t} b_{q_t}(O_t)$$
   dengan $Q = q_1, q_2, q_3, \ldots, q_T$. Terlihat orde perkalian tersebut adalah $(2TN^T)$. Sebagai ilustrasi, dari sebuah sinyal suara dengan durasi 1 detik yang disampling dengan frekuensi 1,28 kHz dan dibaca per frame 30 ms dengan overlap antar frame 40% akan diperoleh T sebesar 71. Hal ini berarti jumlah komputasi untuk menghitung peluang observasi dari sebuah suara yang hanya 1.28 detik dengan model HMM dengan tiga *hidden state* adalah kurang lebih sebesar $2*71*3^{71}$, sehingga diperlukan algoritme yang efisien. Ada dua algoritme yang bisa diterapkan, yaitu algoritme *Forward* dan *Backward*, **[20]**, dengan kompleksitas $O(2N^2T)$. Untuk kasus di atas, jumlah komputasi hanya sekitar $2*3^2*71$ atau 1278.
2. **Problem 2 (Decoding) :** Proses *decoding* dari model $\lambda = (A,B,Л)$ adalah memilih barisan state $Q = q_1, q_2, \ldots, q_T$

yang 'optimal', yaitu yang paling besar kemungkinannya menghasilkan observasi $O = O_1, O_2, O_3, \ldots, O_T$.

Solusi :

Pada problem 1, solusi diperoleh melalui penjumlahan peluang observasi pada semua kemungkinan barisan *state* yang bisa terjadi, sehingga solusi yang diberikan bersifat pasti. Sedangkan pada problem 2, solusi tergantung dari kriteria optimum yang dipakai. Ada beberapa kriteria optimum, yaitu :

a. Memaksimumkan banyaknya *hidden state* yang sesuai. Besaran untuk optimisasi ini adalah $\gamma_t(i) = P(q_t = S_i \mid O, \lambda)$ yaitu peluang pada periode t, *state* yang muncul adalah S_i kalau diketahui observasinya adalah O, dan dirumuskan sebagai :

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(O \mid \lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)}$$

Hidden state yang paling mungkin untuk setiap periode t adalah :

$$q_t = \arg\max_{1 \le i \le N} \gamma_t(i) \qquad \text{untuk } 1 \le t \le T$$

Kelemahan algoritme ini adalah tidak memperhatikan adanya transisi *state* yang tidak mungkin, sehingga tidak menutup kemungkinan munculnya barisan *state* yang 'janggal'.

b. Modifikasi kriteria point a, yaitu dengan memaksimumkan banyaknya segmen hidden state yang benar, yaitu dua (qt,qt+1), tiga (qt,qt+1,qt+2), atau lebih segmen hidden state yang berurutan.

c. Menemukan satu barisan hidden state (path) yang paling sesuai. Solusi dengan kriteria ini diperoleh dengan memaksimumkan peluang kemunculan barisan state, Q, untuk O dan λ tertentu, $P(Q|O,\lambda)$, yang setara dengan memaksimumkan $P(O,Q|\lambda)$. Pencarian path ini dilakukan dengan konsep pemrograman dinamik (dynamic programming) dan dikenal dengan algoritme viterbi.

3. Problem 3 (Learning) : Problem 3 ini adalah berkaitan dengan pembelajaran model HMM dengan menggunakan data yang ada, yang pada dasarnya adalah melakukan pendugaan terhadap parameter model HMM, yaitu A, B dan Л. Seperti pada proses pendugaan parameter pada umumnya, hasil pendugaan tergantung kriteria optimum yang dipakai.

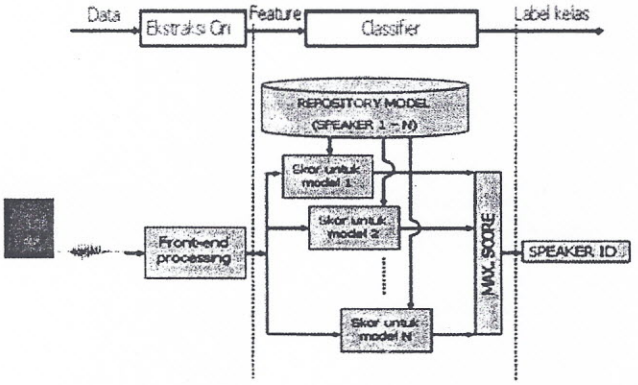
Solusi :

Yang menjadi tujuan dari pembelajaran adalah menentukan parameter model HMM dari suatu set data, sedemikian sehingga model mampu mengenali obyek baru yang mempunyai karakteristik "mirip" dengan data yang dipergunakan untuk training tersebut. Secara umum sudah dikenal dua jenis pendugaan parameter HMM, yaitu teknik Segmental K-Means yang menggunakan kriteria memaksimumkan $P(O,Q|\lambda)$. Jadi dalam hal ini, dari observasi O, parameter HMM diarahkan sehingga peluang kemunculan observasi O dan barisan *state* Q tersebut maksimum. Teknik yang kedua adalah Baum-Welch yang kriterianya adalah memaksimumkan peluang kemunculan barisan observasi O, yaitu $P(O|\lambda)$, **[21]**.

Algoritme detail dari ketiga problem tersebut dapat dilihat pada **[20]** dan **[19]**.

3. Power Spektrum vs Bispektrum pada SIP

Sistem identifikasi pembicara (SIP) merupakan proses untuk menentukan secara otomatis siapa pemilik dari suara yang diberikan ke dalam sistem. Blok diagram dari sistem identifikasi pembicara adalah seperti disajikan pada Gambar 16. Pada sistem tersebut seorang pembicara yang akan diidentifikasi berdasarkan suaranya mengucapkan suatu kata atau frase tertentu. Berikutnya pada bagian ekstraksi ciri dihitung nilai ciri (*feature*) dari sinyal suara masukan. Nilai ciri inilah yang diproses di bagian pengenal (*classifier*) untuk diberikan skor sesuai kelas yang telah ada dalam sistem. Sistem akan memberikan label kelas dari sinyal suara masukan tersebut sesuai skor tertinggi.



Gambar 16. Blok Diagram Sistem Identifikasi Pembicara

Input dari sistem tersebut adalah sinyal suara yang berupa gelombang. Pada bagian ini, dilakukan dijitasi energi suara yang berupa gelombang analog untuk menghasilkan sinyal dijital, dengan cara sampling, dilanjutkan kuantisasi dan coding. Setelah diperoleh nilai dijital sinyal suara masukan, maka sebelum masuk ke ekstraksi ciri, dilakukan penghapusan bagian silence pada sinyal tersebut, lalu dibaca dari frame ke frame dengan panjang tertentu dan saling overlap. Kepada setiap frame ini dilakukan proses windowing dengan fungsi window

tertentu, dan dilanjutkan dengan proses ekstraksi ciri, dan akhirnya dikenali.

Metodologi ektraksi ciri yang digunakan menggunakan teknik MFCC dengan 13 koefisien dan nilai inputnya adalah power spektrum dan bispektrum. Sedangkan untuk pengenalan digunakan left-right HMM dengan 3 hidden state.

3.1 Statistik Orde Tinggi

Kalau {x(t)} merupakan suatu barisan dengan rataan nol, maka autokorelasi orde n (atau moment ke n) dari barisan tersebut dirumuskan sebagai :

$$R_n^x(\tau_1,\tau_2,...,\tau_{n-1}) = E\{x(t)x(t+\tau_1)x(t+\tau_2)...x(t+\tau_{n-1})\}$$

Jika x(t) adalah deterministik dan bersifat periodik dengan periode T (artinya x(t)=x(t+T)) serta merupakan proses ergodic, maka nilai ekspektasi tersebut dapat dirumuskan sebagai :

$$E\{x(t)x(t+\tau_1)x(t+\tau_2)...x(t+\tau_{n-1})\} = \lim_{T\to\infty}\frac{1}{T}\int_0^T x(t)x(t+\tau_1)x(t+\tau_2)...x(t+\tau_{n-1})dt$$

Spektrum orde ke n yang disimbolkan dengan $C_n^x(\omega_1,\omega_2,...,\omega_{n-1})$ dari proses {x(t)} didefinisikan sebagai transformasi Fourier dari moment ke n.

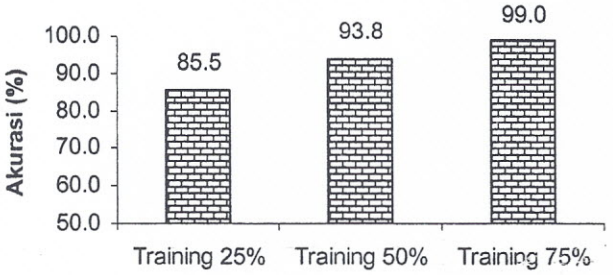
$$C_n^x(\omega_1,\omega_2,...,\omega_{n-1}) = \int_{-\infty}^{\infty}...\int_{-\infty}^{\infty} R_n^x(\tau_1,\tau_2,...,\tau_{n-1})e^{\left\{-j\sum_{i=1}^{n-1}\omega_i\tau_i\right\}}d\tau_1 d\tau_2...d\tau_{n-1}$$

Spektrum orde 2, 3 dan 4, masing-masing disebut sebagai Power spektrum, Bispektrum dan Trispektrum. Bahasan detail mengenai hal ini dapat dilihat di **[22]**.

3.2 Hasil Percobaan

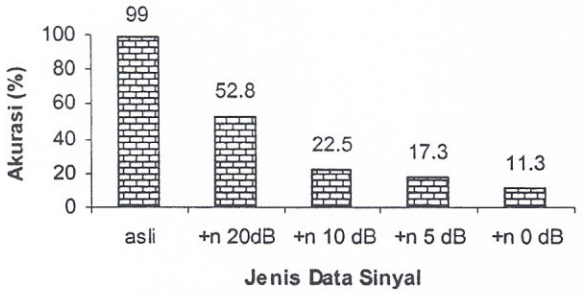
Penelitian ini menggunakan data dari 10 pembicara yang mengucapkan ujaran "pudesha" tanpa pengkondisian masing-masing sebanyak 80 kali yang disampling dengan frekuensi 1.1 kHz. Proporsi data training yang dicobakan adalah 75%, 50% dan 25%. Berikutnya, dibuat lima set data uji, yaitu sinyal asli dan sinyal asli dengan penambahan noise (20 dB, 10 dB, 5 dB, dan 0 dB). Proses ekstraksi ciri menggunakan metode MFCC baik untuk power spektrum maupun bispektrum. Tahapan proses detail dapat dilihat pada [6].

Gambar 17 menyajikan perbandingan hasil akurasi dari sistem dengan power spektrum untuk 3 proporsi data training. Terlihat bahwa dengan jumlah data training meningat (25%, 50%, dan 75%), maka akurasi sistem meningkat dari 85% menjadi 99%. Hal ini mengatakan bahwa untuk pengembangan model, diperlukan jumlah data training yang memadai, yaitu sekitar 60 contoh.



Gambar 17. Perbandingan Akurasi Sistem Berbasis Power Spektrum untuk Sinyal Asli

Namun demikian, untuk data uji yang sudah ditambah noise, terlihat akurasi sistem langsung turun secara nyata, seperti ditunjukkan pada Gambar 18.

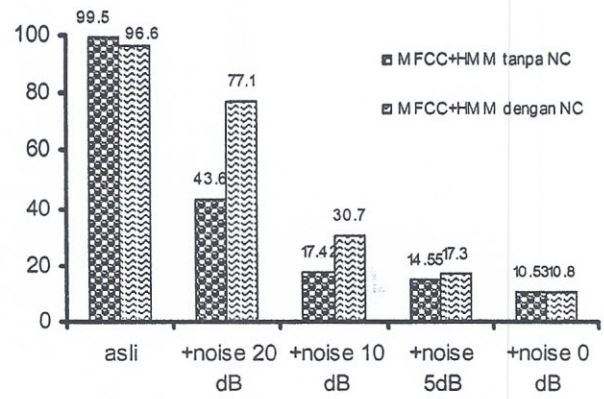


Gambar 18. Perbandingan Akurasi Sistem Berbasis Power Spektrum untuk Sinyal Asli dan dengan Penambahan Noise

Untuk itu dicoba dilakukan perbaikan dengan melakukan proses noise canceling pada sinyal sebelum dilakukan penghitungan power spektrum. Berikutnya dilakukan pengujian kembali, dan hasilnya adalah seperti disajikan pada Gambar 19.

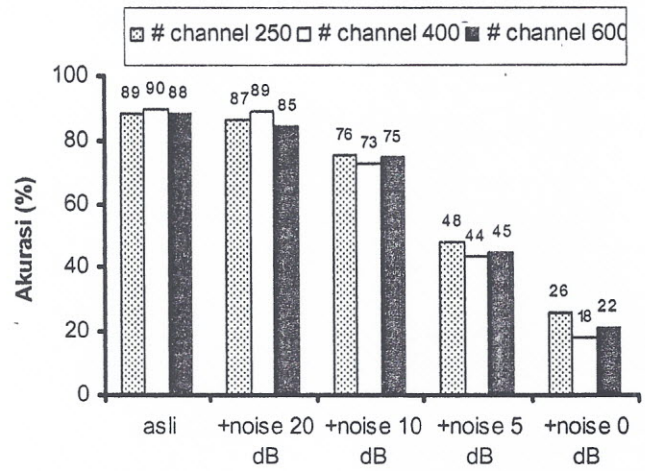
Dari Gambar 19 terlihat bahwa teknik noise canceling mampu meningatkan akurasi sistem pada semua level noise, kecuali pada sinyal asli yang justru menurun dari 99.5% menjadi 96.5%. Hal ini berarti bahwa untuk sinyal asli, penggunaan noise canceling akan mengurangi informasi yang ada pada sinyal, sehingga akurasi sistem turun sekitar 3%. Terlihat noise canceling mampu bekerja hingga noise 20 dB dengan hasil akurasi naik dari 43.6% menjadi 77.1 %. Untuk noise yang lebih besar, teknik ini

gagal bekerja dengan baik, meskipun akurasinya meningkat dibanding tanpa penggunaan noise canceling.



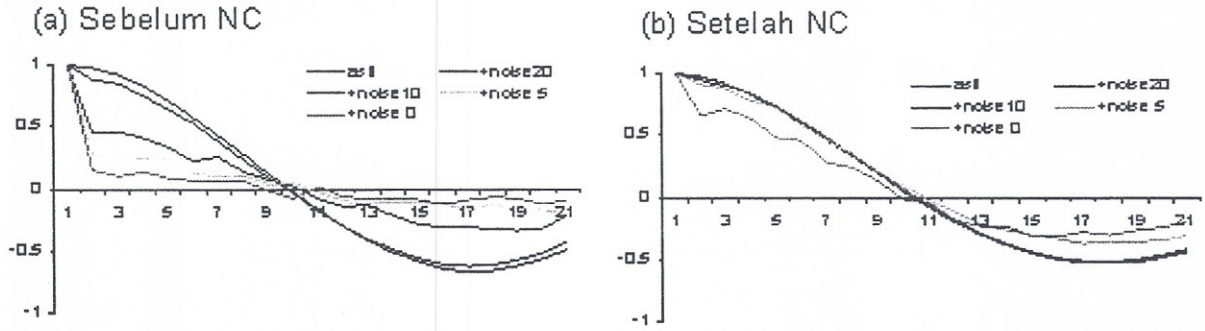
Gambar 19. Perbandingan Akurasi untuk Sistem Berbasis Power Spektrum dengan dan tanpa Noise Cancelling) NC pada Berbagai *Noise*

Untuk melihat bagaimana noise canceling ini berkerja, perhatikan Gambar 20 yang menyajikan pola autokorelasi sinyal asli dengan sinyal yang mengalami penambahan noise. Terlihat pada semua level noise, nilai autokorelasi



Gambar 21. Perbandingan Akurasi Sistem Berbasis Bispektrum dengan Praproses Kuantisasi Vektor

Dari Gambar 22 terlihat bahwa teknik rata-rata di atas kuartil ke tiga secara relatif memberikan hasil yang lebih baik dibanding tiga teknik lainnya. Hal ini terjadi pada semua jenis sinyal, mulai dari sinyal asli, dan sinyal dengan penambahan noise dari 20 dB hingga 0 dB.

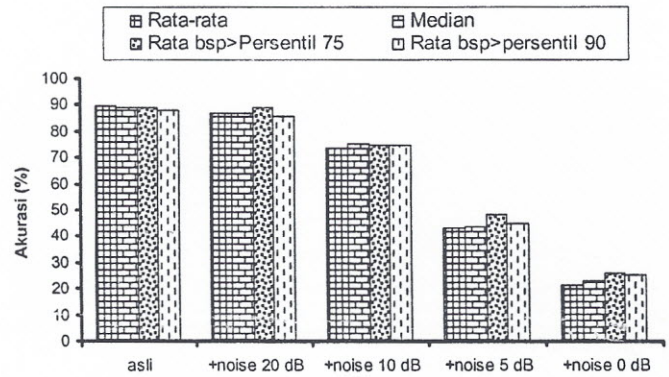


Gambar 20. Autokorelasi Sinyal Suara antara Sebelum dan Setelah ANC

menjadi lebih besar seperti pada sinyal asli.

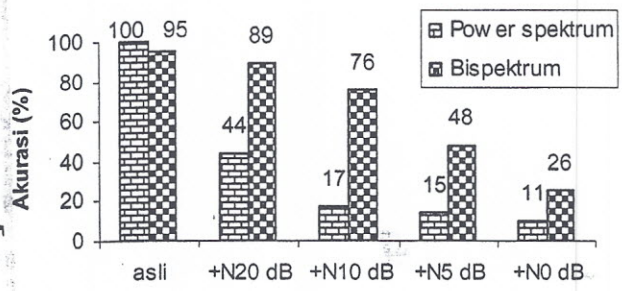
Pada Gambar 21 disajikan hasil akurasi sistem dengan bispektrum yang dikuantisasi vektor dengan berbagai jumlah channel. Secara untuk terlihat bahwa penurunan akurasi tidak terjadi secara gradual sesuai dengan meningkatnya noise. Namun demikian untuk sinyal asli akurasi hanya berkisar 90%, yang hampir sama dengan untuk sinyal bernois 20dB.

Untuk mengetahui teknik yang baik dalam menduga bispektrum pada setiap channel, maka dicobakan empat cara, yaitu nilai rata-rata, nilai median, rata-rata di atas kuartil 3, dan rata-rata di atas persentil 90. Gambar 22 menyajikan perbandingan akurasi untuk sistem berbasis bispektrum dari 4 cara menduga nilai bispektrum pada setiap channel.



Gambar 22. Perbandingan Akurasi antar Berbagai Jenis Statistik pada Setiap Channel

Untuk melihat efektifitas kedua teknik, yaitu power spektrum dan bispektrum, perhatikan Gambar 23 yang menyajikan perbandingan akurasi dari kedua metode tersebut.



Gambar 23. Perbandingan Akurasi antara Sistem Berbasis Power Spektrum dan Bispektrum pada Berbagai level noise

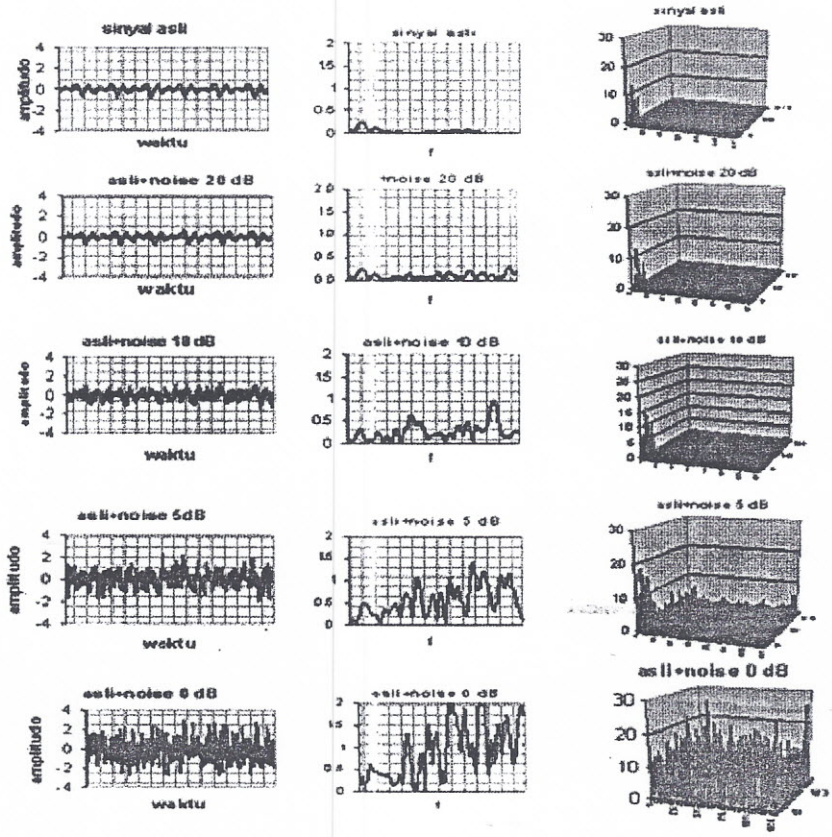
Dari Gambar 23 terlihat secara jelas bahwa bispektrum mampu menghasilkan sistem yang jauh lebih robust terhadap noise dibanding dengan power spektrum. Namun demikian untuk sinyal asli, akurasi sistem dengan bispektrum hanya 95%, sedangkan untuk power spektrum mencapai 100%. Hal ini menunjukkan bahwa teknik kuantisasi yang ada masih perlu ditingkatkan kembali.

Untuk melihat efektifitas bispektrum dalam menekan pengaruh noise, perhatikan Gambar 24 yang menyajikan pola power spektrum dan bispketrum pada berbagai jenis sinyal. Terlihat bahwa nilai power spektrum mengalami distorsi yang nyata, terutama untuk sinyal dengan penambahan noise mulai 10 dB. Untuk noise 20 dB, terlihat perubahan hanya masalah nilai, bukan bentuk umumnya. Hal inilah yang menjadi alasan, mengapa untuk noise 20 dB, teknik noise canceling masih bisa bekerja. Untuk nilai bispektrum, terlihat bahwa pola maupun nilainya relatif tidak berubah hingga noise di atas 0 dB. Perubahan nyata nampak untuk noise 0 dB. Pada noise 0 dB, terlihat bahwa teknik bispektrum sudah tidak bersifat kekar lagi terhadap noise.

4. Kesimpulan

Dari pembahasan di atas dapat ditarik beberapa kesimpulan, di antaranya adalah :

1. Riset dibidang sistem pemrosesan suara masih terbuka luas, baik dari aspek pemrosesan sinyal maupun pemodelan bahasa. Trend metode yang dikembangkan ke arah pemodelan statistik yang merupakan pengembangan dari HMM.



Gambar 24 Perbadingan Pola Power Spektrum (tengah) dan Bispketrum (paling kanan) pada Berbagai Jenis Sinyal (paling kiri)

2. Permasalahan dari pemrosesan sinya suara adalah pada noise dan intervariability pada pembicara.
3. Statistik power spektrum mampu menangkap ciri sinyal dengan baik, sehingga sistem yang dihasilkan mencapai akurasi 99%. Namun demikian, bersifat sensitif terhadap noise. Dengan noise 20 dB saja nilai statistik ini mengalami perubahan yang signifikan, sehingga sistem yang dihasilkan turun drastis. Teknik penghapusan noise tidak mampu memperbaiki kinerja sistem dengan baik.
4. Statistik bisrpektrum mampu memperbaiki kekurangan ini, sehingga akurasinya relatif lebih tinggi dibanding sistem dengan power spektrum pada semua level noise. Namun demikian, dimensi bispektrum adalah tinggi, sehingga diperlukan waktu proses yang lebih lama.

4. Kesimpulan

Dari pembahasan di atas dapat ditarik beberapa kesimpulan, di antaranya adalah :

5. Riset dibidang sistem pemrosesan suara masih terbuka luas, baik dari aspek pemrosesan sinyal maupun pemodelan bahasa. Trend metode yang dikembangkan ke arah pemodelan statistik.
6. Permasalahan dari pemrosesan sinya suara adalah pada noise dan intervariability pada pembicara.
7. Statistik power spektrum mampu menangkap ciri sinyal dengan baik, sehingga sistem yang dihasilkan mencapai akurasi 99%. Namun demikian, bersifat sensitif terhadap noise. Dengan noise 20 dB saja nilai statistik ini mengalami perubahan yang signifikan, sehingga sistem yang dihasilkan turun drastis. Teknik penghapusan noise tidak mampu memperbaiki kinerja sistem dengan baik.
8. Statistik bisrpektrum mampu memperbaiki kekurangan ini, sehingga akurasinya relatif lebih tinggi dibanding sistem dengan power spektrum pada semua level noise. Namun demikian, dimensi bispektrum adalah tinggi, sehingga diperlukan waktu proses yang lebih lama.

REFERENSI

[1] Jurafsky dan J. H. Martin, 2000. *Speech and* Language Processing : An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Prentice Hall Inc., New Jersey.

[2] Deng, Li. Expanding the Scope of Signal Processing. IEEE Signal Pr, May, 2008. ocessing Magazine, Vol. 25, No. 3, hal 2-4.

[3] Joseph P Camphell, September 1997. Speaker Recognition : A Tutorial. Proceeding of the IEEE, Vol. 85, No. 9, hal 1437 - 1460.

[4] J.S. Carmona, 1995. A Hybrid System with Symbolic AI and Statistical Methods for Speech Recognition. Thesis, University of Washington, Washington.

[5] Reynolds, D., 2002. Automatic Speaker Recognition Acoustics and Beyond. Tutorial note, MIT Lincoln Laboratory.

[6] Buono, A., W. Jatmiko, and B. Kusumoputro, 2009. Representasi Nilai HOS dan Model MFCC sebagai Ekstraksi Ciri pada Sistem Identifikasi Pembicara di Lingkungan Ber-noise Menggunakan HMM. Disertasi Program Doktor Ilmu Komputer Fakultas Ilmu Komputer Universitas Indonesia.

[7] Furui, S., 1997. Recent Advances in Speaker Recognition. Pattern Recognition Letters 18, Elsevier.

[8] Todor D. Ganchev, 2005. Speaker Recognition. PhD Dissertation, Wire Communications Laboratory, Department of Computer and Electrical Engineering, University of Patras Greece.

[9] Al-Akaidi, M., 2007. Fractal Speech Processing. Cambridge University Press.

[10] Proakis, J.G., dan D.G. Manolakis, 1996. Digital Signal Processing : Principles, Algorithm, and Applications. Edisi ke tiga, Prentice Hall, New Jersey.

[11] Pieczynski, W. Pairwise Markov, May 2003. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 5.

[12] Derrode, S. Dan W. Pieczynski, September 2004. Signal and Image Segmentation Using Pairwise Markov Chains. IEEE Transactions on Signal Processing, Vol. 52, No. 9, hal 2477-2489.

[13] Pawar, R. V., P. P. Kajane, dan S. N. Mali, August 2005. Speaker Identification Using Neural Networks. Transactions on Engineering, Computing and Technology. V7, ISSN 1305-5313, hal 429-433.

[14] Mohamed, M. A. dan P. Gader, February 2000.. Generalized Hidden Markov Models-Part I: Theoritical Frameworks. IEEE Transactions on Fuzzy Systems, Vol. 8, No. 1, hal 67-81.

[15] Mohamed, M. A. dan P. Gader, February 2000. Generalized Hidden Markov Models-Part II: Application to Handwritten Word Recognition. IEEE Transactions on Fuzzy Systems, Vol. 8, No. 1, hal 82-94.

[16] Hosseyndoost, F. dan M. Teshnehlab, June 2005. Phoneme Classification and Phonetic Transcription Using a New Fuzzy Hidden Markov Model. WSEAS Transaction on Computers, Issue 6, Vol. 4, 541-547.

[17] Pan, H., S. E. Levinson, dan T. Z. Liang, March 2004. A Fused Hidden Markov Model with Application to Bimodal Speech Processing. IEEE Transactions on Signal Processing, Vol. 52, No. 3, hal 537-581.

[18] Othman, H. dan T. Aboulnasr, October 2003. A Separable Low Complexity 2D HMM with Application to Face Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 25, No. 10.

[19] Dugad, R. Dan U.B. Desai, 1996 A Tutorial on Hidden Markov Model. Technical Report, Departement of

Electrical Engineering, Indian Institute of Technology, Bombay.
[20] L. Rabiner, February 1989. A Tutorial on Hidden Markov Model and Selected Applications in Speech Recognition. Proceeding IEEE, Vol 77 No. 2.
[21] Duda, R.O., P.E. Hart, dan D.G. Stork, 2001. Pattern Classification. Edisi Ke dua, John Wiley & Sons, INC.

Agus Buono, memperoleh gelar Sarjana dan Master bidang statistik di IPB pada tahun 1992 dan 1996. Gelar Master dan Doktor bidang Ilmu Komputer diperoleh dari Universitas Indonesia pada tahun 2000 dan 2009. Saat ini sebagai Staf Pengajar Departemen Ilmu Komputer Institut Pertanian Bogor.

Benyamin Kusumoputro, memperoleh gelar Sarjana bidang fisika dari Institut Teknologi bandung dan Doktor Optoelektronika dari Tokyo Institute of Technology-Jepang. Gelar Profesor diperoleh pada tahun 2002 dari Universitas Indonesia. Saat ini sebagai Staf Pengajar Fakultas Teknik Universitas Indonesia.

Wisnu Jatmiko, memperoleh gelar Sarjana Elektro dan Magister Ilmu Komputer dari Universitas Indonesia. Ph.D bidang komputer diperoleh dari Jepang pada tahun 2008. Saat ini sebagai Dosen Fakultas Ilmu Komputer Universitas Indonesia.