



**TESIS - KS142501
VERIFIKASI TANDA TANGAN DINAMIS MENGGUNAKAN FITUR
HISTOGRAM DAN ALGORITMA DYNAMIC TIME WARPING**

**ANO RANGGA RAHARDIKA
05211450010018**

**DOSEN PEMBIMBING
Dr. Ir. Aris Tjahyanto, M.Kom.
NIP. 196503101991021001**

**PROGRAM MAGISTER
DEPARTEMEN SISTEM INFORMASI
FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2018**



THESES - KS142501
Dynamic Signature Verification Using Histogram Features and
Dynamic Time Warping Algorithm

ANO RANGGA RAHARDIKA
05211450010018

SUPERVISOR
Dr. Ir. Aris Tjahyanto, M.Kom.
NIP. 196503101991021001

MAGISTER PROGRAM
DEPARTMENT IN INFORMATION SYSTEM
FACULTY OF INFORMATION TECHNOLOGY AND COMMUNICATION
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2018

LEMBAR PENGESAHAN TESIS

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Komputer (M.Kom)

di
INSTITUT TEKNOLOGI SEPULUH NOPEMBER

Oleh:

Ano Ranga Rahardika
NRP. 05211450010018

Tanggal Ujian : 12 Juli 2018
Periode Wisuda : September 2018

Disetujui Oleh:

Dr. Ir. Aris Tjahyanto, M.Kom.
NIP. 19650310 19910 2 1001

Dr. Eng. Febriliyan Samopa, S.Kom., M.Kom.
NIP. 19730219 19980 2 1001

Erma Suryani, S.T., M.T. Ph.D
NIP. 19700427 20050 1 2001



(Pembimbing)



(Penguji I)



(Penguji II)

Dekan
Fakultas Teknologi Informasi dan Komunikasi



Dr. H. Agus Zainal Arifin, S.Kom., M.Kom
NIP. 19720809 19951 2 1001

Verifikasi Tanda Tangan Dinamis Menggunakan Fitur Histogram dan Algoritma Dynamic Time Warping

Nama Mahasiswa : Ano Ranga Rahardika
NRP : 05211450010018
Pembimbing : Dr. Ir. Aris Tjahyanto, M.Kom.

ABSTRAK

Sejauh ini, fitur-fitur histogram telah mampu memberikan hasil yang cukup baik untuk verifikasi tanda tangan dinamis dengan menggunakan metode pengukuran jarak *manhattan distance*. Berdasarkan studi literatur ada kemungkinan untuk meningkatkan kinerja sistem. Yang pertama adalah penggunaan *Dynamic Time Warping* (DTW) sebagai pengganti *manhattan distance*. Hal tersebut karena DTW merupakan algoritma yang sudah dikenal dengan baik pada penelitian verifikasi tanda tangan dinamis sebelumnya. Perbedaannya adalah DTW biasanya digunakan langsung pada sinyal posisi X dan Y tanpa melalui ekstraksi fitur histogram. Yang kedua adalah penambahan fitur-fitur baru dari histogram X dan Y karena histogram X dan Y adalah gambaran distribusi posisi x dan y yang berkaitan dengan “bentuk” tanda tangan yang merupakan salah satu unsur penting dalam kinerja verifikasi tanda tangan dinamis. Untuk mengukur kinerja sistem, digunakan nilai AUC (Area Under Curve). Nilai AUC memiliki interpretasi yaitu nilai kemungkinan tanda tangan asli terverifikasi sebagai asli daripada tanda tangan palsu terverifikasi sebagai asli. Sehingga nilai AUC yang lebih besar lebih baik.

Pada pengujian pertama secara garis besar DTW lebih buruk daripada Manhattan Distance. Nilai AUC sistem verifikasi dengan metode DTW selalu lebih kecil daripada sistem verifikasi dengan metode Manhattan Distance. Pada dataset SVC2004 kasus pemalsuan terlatih nilai AUC DTW lebih buruk dengan selisih 0.02623. Pada kasus pemalsuan acak nilai AUC DTW juga lebih buruk dengan selisih 0.00029. Hal yang sama juga didapatkan pada dataset Privat. Pada kasus pemalsuan terlatih nilai AUC DTW lebih kecil dengan selisih 0.053600166. Pada kasus pemalsuan acak nilai AUC DTW lebih kecil dengan selisih 0.047. Pada pengujian kedua menunjukkan kinerja dengan fitur-fitur baru lebih baik. Nilai AUC sistem dengan fitur-fitur baru selalu lebih baik atau lebih besar. Pada dataset SVC2004 kasus pemalsuan terlatih nilai AUC sistem dengan fitur baru lebih besar dengan selisih 0.00654. Sedangkan pada kasus pemalsuan acak nilai AUC sistem dengan fitur baru lebih besar dengan selisih 0.0549. Pada dataset Privat kasus pemalsuan terlatih nilai AUC sistem dengan fitur baru lebih besar dengan selisih 0.024. Pada kasus pemalsuan acak nilai AUC lebih besar dengan selisih 0.08.

Kata kunci: fitur, histogram, pemalsuan terlatih, pemalsuan acak, AUC, Privat, SVC2004, Manhattan, DTW

(lembar ini sengaja dikosongkan)

DYNAMIC SIGNATURE VERIFICATION USING HISTOGRAM FEATURES AND DYNAMIC TIME WARPING ALGORITHM

By : Ano Ranga Rahardika
Student Identity Number : 05211450010018
Supervisor : Dr. Ir. Aris Tjahyanto, M.Kom.

So far, histogram features have been able to provide good results for dynamic signature verification using manhattan distance measurement method. Based on literature studies it is possible to improve system performance. The first is the use of Dynamic Time Warping (DTW) instead of manhattan distance. That's because DTW is a well-known algorithm in previous dynamic signature verification research. The difference is that DTW is usually used directly on X and Y position signals without going through histogram feature extraction. The second is the addition of new features of the histogram of X and Y because the histogram of X and Y is an illustration of the distribution of x and y positions that relate to the "shape" element of the signature which is one of the key elements in the performance of dynamic signature verification. To measure the performance of the system, the value of AUC (Area Under Curve) is used. The AUC value has an interpretation that is the probability value of genuine signature verified as the genuine rather than forgery signature verified as the genuine. So the larger value of AUC is better.

In the first test's result, the DTW was worse than Manhattan Distance. The AUC value of DTW method is always smaller than the AUC value of Manhattan Distance method. In the SVC2004 dataset of the skilled forgery case, the AUC value of DTW is worse by 0.02623 difference. In the case of random forgery the AUC value of DTW is also worse by 0.00029 difference. The same result was also reported in the Privat dataset. In the case of skilled forgery the AUC value of DTW is smaller by 0.053600166 difference. In the case of random forgery the AUC value of DTW is smaller by 0.047 difference. In the second test, it was reported that the performance with the new features was better. The AUC value of system with the new features is always greater. In the SVC2004 dataset in the skilled forgery case the AUC value of system with the new features was larger by 0.00654 difference. Whereas in the case of random forgery, the value of AUC system with new feature was larger by 0.0549 difference. In the Private datasets in the skilled forgery case, the AUC value of system with new features was larger by 0.024 difference. In the case of random forgery the AUC value was larger by 0.08 difference

Keywords : feature, histogram, skilled forgery, random forgery, AUC, Privat, SVC2004, Manhattan, DTW

(lembar ini sengaja dikosongkan)

KATA PENGANTAR

Segala puji bagi Allah atas rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan tesis yang berjudul **“Verifikasi Tanda Tangan Dinamis Menggunakan Fitur Histogram dan Algoritma Dynamic Time Warping”** sebagai salah satu syarat kelulusan dari Program Pascasarjana Departemen Sistem Informasi, Fakultas Teknologi Informasi dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya. Dengan terselesaikannya penelitian ini, maka selesai pula masa studi penulis.

Penulis berharap penelitian ini dapat memberikan manfaat bagi pembaca, menjadi salah satu batu loncatan bagi penulis untuk terus berkarya, dan memberikan sumbangsih bagi ilmu pengetahuan. Proses pengerjaan tesis ini telah banyak mendapatkan bantuan, bimbingan, masukan serta dukungan dari berbagai pihak. Sehingga dalam kesempatan ini, penulis mengucapkan terima kasih kepada:

1. Ibu, Bapak, kakaku dan seluruh keluarga besar yang selalu memberikan do’a, motivasi, semangat serta dukungan setiap saat dan tanpa batas.
2. Bapak Dr. Ir. Aris Tjahyanto, M.Kom., selaku pembimbing yang telah memberikan waktu, motivasi, saran dan imunya selama proses membimbing sehingga tesis ini dapat terselesaikan dengan baik.
3. Bapak Dr. Eng. Febriliyan Samopa, S.Kom., M.Kom., selaku dosen penguji I dan Ibu Erma Suryani, S.T, M.T. Ph.D., selaku penguji II yang telah banyak memberikan masukan dan saran dalam perbaikan tesis ini.
4. Karyawan dan Karyawati Tata Usaha Sistem Informasi yang telah meluangkan waktu untuk memberikan data uji berupa tanda tangan dengan penuh kesabaran
5. Keluarga besar PT ARM Solusi, yang telah memberikan motivasi dan kesempatan waktu untuk belajar di tengah pekerjaan yang dilakukan.
6. Rekan-rekan seperjuangan kuliah program Magister Departemen Sistem Informasi angkatan 2014 yang masih bertahan sampai sekarang.
7. Dosen dan Karyawan/wati Institut Teknologi Sepuluh Nopember Surabaya.

Dan berbagai pihak lain yang tidak dapat penulis cantumkan namanya satu per satu yang selama penelitian ini telah memberikan dukungan, bantuan, dan jasanya

terhadap penulis. Semoga Allah SWT., membalas semua kebaikan yang telah dilakukan

Penulis menyadari bahwa tesis ini masih memiliki kekurangan dan ketidaksempurnaan dalam penulisan. Oleh karena itu, penulis mengharapkan kritik dan saran dari pembaca yang bersifat membangun sebagai bahan acuan penelitian selanjutnya. Akhir kata, semoga penelitian ini dapat memberikan manfaat bagi kita semua.

Surabaya, 30 Juli 2018

Penulis

DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN TESIS.....	Error! Bookmark not defined.
ABSTRAK.....	v
KATA PENGANTAR.....	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR.....	xv
DAFTAR TABEL.....	xvii
DAFTAR LAMPIRAN.....	xix
BAB 1 PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Rumusan Masalah.....	6
1.3. Tujuan Penelitian.....	7
1.4. Ruang Lingkup Penelitian.....	7
1.5. Kontribusi Penelitian.....	7
1.6. Sistematika Penulisan Dokumen.....	8
BAB 2 LANDASAN TEORI DAN KAJIAN PUSTAKA.....	11
2.1. Kedinamisan Tanda Tangan.....	11
2.1.1. Kecepatan dan Akselerasi Tanda Tangan.....	13
2.1.2. Contoh Pemalsuan.....	16
2.2. Fitur Tanda Tangan Dinamis.....	19
2.3. Fitur Parameter dari Metode Statistik.....	20
2.4. Histogram.....	23
2.4.1. Lebar dan Jumlah bin.....	24
2.4.2. Fitur Histogram.....	25
2.5. Dataset Tanda Tangan Publik.....	27
2.3.1. Dataset SVC2004.....	27
2.6. Template Tanda Tangan.....	28
2.7. Dynamic Time Warping (DTW).....	29
2.7.1. DTW Klasik.....	30
2.7.2. Fast DTW.....	34
2.8. Kategori Pencocokan Fitur.....	37
2.8.1. Pendekatan Berbasis Jarak.....	38

2.8.2.	Pendekatan Berbasis Model	40
2.9.	Pengukuran Performa	42
2.9.1	Accuracy	43
2.9.2	Precision dan Recall	43
2.9.3	F-Measure	44
2.9.4.	Precision-Recall Curve	44
2.9.5.	<i>Area Under Curve (AUC)</i>	45
2.10.	Pemeringkatan Fitur Berbasis Filter	46
2.10.1.	Information Gain	47
2.10.2.	Gain Ratio	48
2.10.3.	Relief	48
2.10.4.	Correlation	49
BAB 3 METODOLOGI PENELITIAN		51
3.1.	Penyiapan Data	51
3.1.1.	Dataset Privat	51
3.1.2.	Dataset SVC2004	54
3.2	Praproses Data	54
3.2.1.	Normalisasi Waktu	55
3.2.2.	Penggabungan Garis	55
3.3.	Derivasi Data	56
3.4.	Ekstraksi Fitur Histogram	57
3.5.	Pembuatan Template Tanda Tangan Pengguna	60
3.6.	Skenario Uji Coba	61
3.6.1	Uji Coba Perbandingan <i>Dynamic Time Warping</i> dan <i>Manhattan Distance</i>	62
3.6.2.	Uji Coba Perbandingan <i>Kinerja</i> Dengan dan Tanpa Penambahan Fitur Baru	63
3.6.4	Uji Coba Peringkat Fitur	64
3.7.	Analisis Hasil	64
3.8.	Penyusunan Kesimpulan dan Saran Pengembangan Penelitian	64
BAB 4 HASIL DAN PEMBAHASAN		65
4.1.	Data Uji Coba	65
4.2.	Perangkat Uji Coba	65
4.3.	Pelaksanaan dan Hasil Uji Coba	67
4.3.1	Perbandingan <i>Dynamic Time Warping</i> dan <i>Manhattan Distance</i>	68

4.3.2	Perbandingan Kinerja Fitur Baru	76
4.3.4.	Pemeringkatan Fitur.....	85
4.4.	Analisis Hasil Uji Coba.....	88
4.4.1.	Perbandingan Dynamic Time Warping dan Manhattan Distance	89
4.4.2.	Perbandingan Kinerja Fitur Baru	91
4.4.3.	Pemeringkatan Fitur	93
BAB 5 KESIMPULAN DAN SARAN.....		95
5.1.	Kesimpulan.....	95
5.2.	Saran	97
DAFTAR PUSTAKA		99
LAMPIRAN A		106
A.1	Perbandingan Dynamic Time Warping dan Manhattan Distance	106
A.1.1	Kinerja DTW pada Dataset SVC2004	106
A.1.2	Kinerja Manhattan Distance pada Dataset SVC2004	108
A.1.3	Kinerja DTW pada Dataset Privat.....	110
A.1.4	Kinerja Manhattan Distance pada Dataset Privat.....	112
A.2	Perbandingan Kinerja Fitur Baru	114
A.2.1	Kinerja Dengan penambahan Fitur Baru pada Dataset SVC2004	114
A.2.2	Kinerja Dengan Penambahan Fitur Baru pada Dataset Privat	118
BIOGRAFI PENULIS		123

(lembar ini sengaja dikosongkan)

DAFTAR GAMBAR

Gambar	Halaman
2.1 Grafik x dan y dari tanda tangan	12
2.2 Kecepatan x dan y dari tanda tangan menggunakan Forward Differences	14
2.3 Kecepatan x dan y dari tanda tangan menggunakan Central Differences..	15
2.4 Percepatan x dan y dari tanda tangan menggunakan Forward Differences	15
2.5 Percepatan x dan y dari tanda tangan menggunakan Central Differences.	16
2.6 x dan y dari tanda tangan palsu	17
2.7 Kecepatan x dan y tanda tangan palsu menggunakan Central Differences	17
2.8 Percepatan x dan y tanda tangan palsu menggunakan Central Differences	18
2.9 Kategori Fitur Tanda Tangan	19
2.10 Pengambilan Fitur Parameter Statistik	20
2.11 Contoh Histogram 1 Dimensi	24
2.12 Pensejajaran waktu dari dua deret. Titik yang disejajarkan ditunjukkan dengan panah	30
2.13 Cost Matrix dari dua deret X (vertikal aksis) dan Y (horizontal aksis) ...	31
2.14 Ilustrasi path dari pasangan indeks dari deret X dengan panjang N=9 dan deret Y dengan panjang M=7	32
2.15 Empat resolusi berbeda yang dievaluasi selama berjalannya algoritma FastDTW	36
2.16 Kategori Pencocokan Fitur	37
2.17 Alur pendekatan Berbasis Model	42
2.18 Kurva Precision-Recall	45
2.19 Area Under Curve (AUC)	46
3.1 Metodologi Penelitian	52
3.2 Proses Enrollement Data Tanda Tangan	53
3.3 Halaman unduh dataset SVC2004	54
3.4 Dekomposisi tandan tangan dinamis menjadi <i>time series x</i> dan <i>time series y</i>	56
3.5 Contoh Histogram X digambarkan dalam bentuk grafik	59
3.6 Proses Verifikasi Skenario Perbandingan DTW dan Manhattan	63
3.7 Proses Verifikasi Skenario Perbandingan Dengan dan Tanpa Fitur Baru .	64
4.1 Kurva Precision-Recall metode DTW dalam dataset SVC2004	69
4.2 AUC metode DTW dalam dataset SVC2004	69
4.3 Kurva Precision-Recall metode Manhattan dalam dataset SVC2004	71
4.4 AUC metode Manhattan dalam dataset SVC2004	71
4.5 Kurva Precision-Recall metode DTW dalam dataset Privat	73
4.6 AUC metode DTW dalam dataset Privat	73
4.7 Kurva Precision-Recall metode Manhattan dalam dataset Privat	75
4.8 AUC metode Manhattan dalam dataset Privat	75
4.9 Kurva Precision-Recall Semua Fitur Baru dalam dataset SVC2004	77
4.10 AUC Semua Fitur Baru Dataset SVC2004	77
4.11 AUC Dengan 12 Fitur Baru peringkat Tertinggi Dataset SVC2004	79
4.12 ROC Dengan 12 Fitur Baru peringkat Tertinggi Dataset SVC2004	79

4.13 Kurva Precision-Recall Semua Fitur Baru dalam dataset Privat	82
4.14 AUC Dengan Fitur Baru Dataset Privat	82
4.15 AUC Dengan 12 Fitur Baru peringkat Tertinggi Dataset Privat	84
4.16 Kurva Precision-Recall Dengan 12 Fitur Baru peringkat Tertinggi Dataset Privat	85
4.17 Peringkat 12 fitur baru Dataset SVC2004 dengan tiga metode yang berbeda dalam bentuk grafik	86
4.18 Peringkat 12 fitur baru Dataset Privat dengan tiga metode yang berbeda dalam bentuk grafik	87

DAFTAR TABEL

Tabel	Halaman
2.1 Penelitian Terkait Fitur Global Statistik	22
2.2 Confusion Matrix	42
3.1 Contoh 5 titik pertama hasil <i>enrollment</i>	53
3.2 Contoh derivasi deret posisi atau Himpunan X.....	57
3.3 Daftar Histogram Asli	57
3.4 Daftar Histogram yang ditambahkan	58
3.5 Contoh Histogram X Dalam Bentuk Tabel.....	59
4.1 Jumlah User Per Dataset	65
4.2 Spesifikasi Perangkat Mobile.....	66
4.3 Spesifikasi Perangkat Keras Notebook Host.....	66
4.4 Spesifikasi Perangkat Keras Mesin Virtualisasi	66
4.5 Spesifikasi Perangkat Lunak	66
4.6 Spesifikasi Tools Pemeringkatan Fitur	67
4.7 Confusion Matrix DTW pada dataset SVC2004.....	68
4.8 Confusion Matrix DTW threshold optimal di dataset SVC2004.....	69
4.9 Confusion Matrix Manhattan pada dataset SVC2004.....	70
4.10 Confusion Matrix Manhattan threshold optimal di dataset SVC2004.....	71
4.11 Confusion Matrix DTW pada dataset Privat.....	72
4.12 Confusion Matrix DTW threshold optimal di dataset Privat	73
4.13 Confusion Matrix Manhattan pada dataset Privat.....	74
4.14 Confusion Matrix Manhattan threshold optimal di dataset Privat	75
4.15 Confusion Matrix Dengan Fitur Baru pada dataset SVC2004.....	76
4.16 Confusion Matrix Fitur Baru Threshold Optimal dataset SVC2004	77
4.17 Peringkat Fitur Berdasarkan Nilai AUC_SF Dataset SVC2004	78
4.18 Nilai AUC_SF dan AUC_RF per kelompok kelipatan 4 serta 13,14, dan 15 besar Dataset SVC2004.....	78
4.19 Confusion Matrix Dengan 12 Fitur Peringkat Tetinggi Dataset SVC2004	79
4.20 Confusion Matrix 12 Fitur Baru Threshold Optimal dataset SVC2004 .	80
4.21 Confusion Matrix Dengan Fitur Baru pada dataset Privat.....	81
4.22 Confusion Matrix Fitur Baru Threshold Optimal dataset Privat.....	82
4.23 Peringkat Fitur Berdasarkan Nilai AUC_SF Dataset Privat.....	83
4.24 Nilai AUC_SF dan AUC_RF per kelompok kelipatan 4 serta 13,14, dan 15 besar Dataset Privat.....	83
4.25 Confusion Matrix Dengan 12 Fitur Peringkat Tetinggi Dataset Privat....	84
4.26 Confusion Matrix 12 Fitur Baru Threshold Optimal dataset Privat.....	85
4.27 Peringkat 12 fitur baru dengan metode yang berbeda Dataset SVC2004	86
4.28 Peringkat 12 fitur baru dengan metode yang berbeda Dataset Privat	87
4.29 Perbandingan Kinerja DTW dan Manhattan pada dataset SVC2004	89
4.30 Perbandingan Kinerja DTW dan Manhattan pada dataset Privat.....	90
4.31 Kinerja pada dataset SVC2004	91
4.32 Kinerja pada dataset Privat	92

(lembar ini sengaja dikosongkan)

DAFTAR LAMPIRAN

Lampiran A.1 : Perbandingan Dynamic Time Warping dan Manhattan Distance

Lampiran A.1.1 : Kinerja DTW pada Dataset SVC2004

Lampiran A.1.2 : Kinerja Manhattan Distance pada Dataset SVC2004

Lampiran A.1.3 : Kinerja DTW pada Dataset Privat

Lampiran A.1.4 : Kinerja Manhattan Distance pada Dataset Privat

Lampiran A.2 : Perbandingan Kinerja Fitur Baru

Lampiran A.2.1 : Kinerja Dengan Penambahan Fitur Baru pada Dataset SVC2004

Lampiran A.2.2 : Kinerja Dengan Penambahan Fitur Baru pada Dataset Privat

(lembar ini sengaja dikosongkan)

BAB 1

PENDAHULUAN

Pada bab awal ini dijelaskan mengenai gambaran penelitian dari latar belakang penelitian, rumusan masalah, tujuan penelitian, batasan penelitian, kontribusi penelitian, hingga sistematika penulisan dokumen.

1.1. Latar Belakang

Beberapa tahun terakhir muncul beberapa penelitian mengenai verifikasi tanda tangan dinamis atau *online*. Salah satu diantaranya adalah penelitian yang dilakukan oleh (Sae-bae & Memon, 2014) yang mengeksplorasi fitur-fitur histogram untuk digunakan dalam verifikasi tanda tangan *online*. Yang perlu menjadi catatan adalah fitur-fitur histogram merupakan fitur yang sering digunakan dalam permasalahan-permasalahan rekognisi atau pengenalan. Sebagai contoh, fitur histogram digunakan dalam permasalahan pengenalan objek (Schiele & Crowley, 1996) dan pengenalan tanda tangan *offline* (Qiao, Liu, & Tanf, 2007). Penggunaan histogram untuk tanda tangan *online* pertama kali diperkenalkan oleh (Nelson, Turin, & Hastie, 1994), dan juga sebagai bagian fitur yang digunakan dalam (Fallah, Jamaati, & Soleamani, 2011; Fierrez-Aguilar, Nanni, Lopez-Peñalba, Ortega-Garcia, & Maltoni, 2005). Namun penggunaan histogram yang mereka gunakan hanya terbatas pada histogram sudut yang diturunkan dari vektor yang menghubungkan dua titik yang bersebelahan. Menurut (Sae-bae & Memon, 2014), lebih banyak informasi yang tersedia dalam tanda tangan dinamis yang dapat digunakan untuk menghasilkan fitur-fitur histogram yang berguna dalam verifikasi tanda tangan *online*. Mereka mencoba menurunkan fitur-fitur histogram yang tidak hanya dari sudut, tetapi juga menurunkan histogram dari perubahan posisi panjang perubahan posisi, perubahan kecepatan, dan panjang perubahan kecepatan dari data tanda tangan dinamis. Selain itu mereka juga membentuk histogram dua dimensi yaitu histogram yang berasal dari gabungan dari dua variabel yang berbeda.

Dalam penelitiannya, (Sae-bae & Memon, 2014) melakukan eksperimen untuk menguji metode yang mereka usulkan terhadap dua dataset tanda tangan *online* yaitu MCYT (Ortega-Garcia et al., 2003) dan SUSIG (Kholmatov &

Yanikoglu, 2009). Template tanda tangan dibangun dengan menggunakan sampel tanda tangan asli sejumlah k pertama yang digunakan untuk data training. Data tanda tangan asli lain yang tersisa, digunakan untuk mengevaluasi True Positive yaitu jumlah tanda tangan asli yang berhasil diverifikasi sebagai asli dan False Negative yaitu jumlah tanda tangan asli yang diverifikasi sebagai palsu. Pemalsuan sendiri dikategorikan menjadi pemalsuan acak (*random forgery*, biasa disingkat RF) dan pemalsuan terlatih (*skilled forgery*, biasa disingkat SF). Contoh kasus pemalsuan dengan kategori pemalsuan acak yaitu ketika penyerang melakukan pemalsuan dengan menggunakan tanda tangannya sendiri. Dalam kasus RF, semua sampel tanda tangan orang lain dapat digunakan untuk data uji. Sedangkan untuk skenario pemalsuan terlatih, sudah disediakan 50 sampel tanda tangan palsu terlatih dari dataset MCYT dan 10 sampel tanda tangan palsu terlatih dari dataset SUSIG untuk masing-masing pengguna. Selain melakukan eksperimen menggunakan dua dataset publik, (Sae-bae & Memon, 2014) juga melakukan eksperimen terhadap dataset yang dibuat menggunakan perangkat mobile yang dimiliki oleh masing-masing pengguna. Pengumpulan tanda tangan pengguna dilakukan selama 6 sesi dalam 7 hari. Semua tanda tangan dikumpulkan menggunakan HTML5. Tujuannya adalah untuk memudahkan pengguna menginput tanda tangan mereka. Algoritma yang digunakan pada dataset mobile sama dengan algoritma yang digunakan saat menggunakan dataset publik kecuali dua hal, pertama: histogram yang berkaitan dengan informasi tekanan tidak digunakan karena informasi tersebut tidak tersedia dalam perangkat mobile. Kedua, ada penambahan langkah *preprocessing* dan perubahan beberapa fitur histogram untuk dataset *mobile*. Setelah melakukan eksperimen, hasilnya menunjukkan bahwa performa verifikasi sistem hanya sedikit lebih rendah dari performa saat menggunakan dataset publik. Kesimpulan yang didapatkan dari penelitian yang mereka lakukan adalah sistem yang mereka proposalkan mampu menghasilkan kinerja yang baik dan mampu mengungguli sistem lain yang menggunakan dataset yang sama.

Walaupun begitu, menurut peneliti hasil tersebut masih kurang maksimal karena dalam penelitian tersebut mereka tidak menggunakan histogram yang diturunkan dari himpunan posisi x dan posisi y (X dan Y), namun yang mereka gunakan adalah histogram yang didapatkan dari $(X^1, Y^1, R^1, \Phi^1, X^2, Y^2, R^2, \Phi^2)$. Dalam

penelitian ini dicoba untuk melakukan penambahan fitur yang diturunkan dari histogram X dan Y , karena histogram X dan Y memberikan informasi tentang distribusi posisi dalam tanda tangan dinamis. Informasi tersebut sangat sensitif terhadap bentuk tanda tangan, apabila bentuk tanda tangan berbeda maka distribusi histogram posisi X dan Y juga akan sangat berbeda. Oleh karena itu, histogram X dan Y handal untuk menangani kasus pemalsuan acak (RF) dan juga pemalsuan terlatih (SF). Selain itu, dalam sesi saran, mereka juga mengusulkan beberapa ide untuk penelitian yang akan datang salah satunya adalah melakukan fusi atau penggabungan dengan pendekatan *Dynamic Time Warping* (DTW) untuk meningkatkan akurasi. DTW atau disebut juga dengan *String Matching* adalah algoritma pencocokan elastis yang telah dikenal dengan baik, dengan mensejajarkan titik sampel dari dua tanda tangan yang memiliki panjang yang berbeda (Sonia Garcia-Salicetti et al., 2009). Eksplorasi awal penggunaan DTW untuk verifikasi tanda tangan online dapat ditemukan di (Jain, Griess, & Connell, 2002), dimana tanda tangan direpresentasikan kedalam fitur lokal (semua titik adalah fitur lokal). DTW mampu mensejajarkan dua vektor dari fitur lokal menggunakan *dynamic programming* dan menghitung nilai ketidaksamaan dari kedua vektor tersebut walaupun panjangnya berbeda. Kemudian (Feng & Wah, 2003) memproposalkan variasi dari DTW yang disebut '*extreme point warping*' dimana hanya titik yang terpilih yang digunakan untuk penghitungan. Selanjutnya (Kholmatov & Yanikoglu, 2005) merepresentasikan tanda tangan dalam bentuk vektor jarak tiga dimensi yang dihitung menggunakan DTW dan model verifikasi dibentuk berdasarkan vektor tiga dimensi ini. Selanjutnya (Yanikoglu & Kholmatov, 2009) juga memproposalkan sistem verifikasi dengan menggabungkan DTW dan deskriptor Fourier. Selain itu, Kombinasi pendekatan kuantisasi vektor dan DTW telah diproposalkan dalam (Faundez-Zanuy, 2007), dimana vektor fitur yang sudah dikuantisasi dari tanda tangan tes, dicocokkan dengan tanda tangan referensi. Oleh karena itu, pada penelitian ini juga mencoba melakukan penggabungan fitur-fitur histogram dengan pendekatan DTW. Fitur-fitur histogram yang diperoleh dari tanda tangan akan dihitung ketidaksamaanya dengan tanda tangan referensi yang sudah disimpan dengan menggunakan DTW. Penggabungan DTW dan fitur global juga pernah dilakukan oleh (Yanikoglu & Kholmatov, 2009)

dimana mereka menggunakan deskriptor fourier sebagai fitur. Penggabungan ini bertujuan untuk meningkatkan kinerja verifikasi tanda tangan *online*, seperti yang disarankan oleh (Sae-bae & Memon, 2014).

Fitur-fitur histogram ini digunakan untuk menangkap karakteristik yang penting dari tanda tangan dan juga menangkap relasi yang terjadi antar fitur yang menjadi ciri khas tanda tangan. Proses ekstraksi fitur yang mereka lakukan dimulai dengan mengkonversi data time-series posisi x dan posisi y menjadi deretan titik kartesian yang disimbolkan dengan X dan Y . Selanjutnya, dari kedua deret atau himpunan tersebut diturunkan menjadi X^1 dan Y^1 untuk turunan pertama dan X^2 dan Y^2 untuk turunan kedua. Dari turunan pertama mereka bisa mendapatkan sudut vektor dari dua pasang titik (x, y) yang berurutan yang disimbolkan dengan Φ^1 dan juga panjang dari vektor tersebut yang disimbolkan dengan R^1 . Dari turunan kedua, mereka bisa mendapatkan sudut vektor dari dua pasang titik (x^1, y^1) yang berurutan yang disimbolkan dengan Φ^2 dan juga panjang dari vektor tersebut yang disimbolkan dengan R^2 . Sehingga secara lengkap deretan atau himpunan yang didapatkan adalah $(X^1, Y^1, R^1, \Phi^1, X^2, Y^2, R^2, \Phi^2)$. Setelah itu, masing - masing deret atau himpunan diubah kedalam bentuk histogram dengan mengelompokkan setiap nilai dalam himpunan tersebut kedalam *bin* dengan lebar yang sudah ditentukan sebelumnya. Jadi, dari satu himpunan akan terbentuk satu histogram yang menggambarkan distribusi nilai dalam himpunan tersebut. Mereka juga mengusulkan pembentukan histogram yang dibuat dari penggabungan dua buah himpunan yang disebut dengan histogram dua dimensi. Tujuannya adalah untuk menangkap hubungan antar himpunan dalam fitur yang dihasilkan. Hasil frekuensi dalam setiap bins di masing-masing histogram itulah yang digunakan sebagai fitur. Setelah tanda tangan selesai direpresentasikan menjadi vektor fitur, langkah selanjutnya adalah pembuatan template referensi. Template tanda tangan pengguna dibuat dengan menggunakan data *training*. Nilai variance setiap komponen fitur digunakan untuk membangun vektor pengkuantisasi. Rata-rata vektor fitur yang sudah dikuantisasi merupakan template untuk setiap pengguna. Vektor pengkuantisasi beserta rata-rata vektor fitur yang sudah dikuantisasi ini disimpan sebagai template identitas pengguna. Saat proses verifikasi berlangsung, fitur-fitur histogram tanda tangan dari pengguna yang mengklaim, akan dibandingkan dengan

template dengan mengukur nilai ketidaksamaan menggunakan *manhattan distance*. Apabila nilai lebih kecil dari *threshold*, maka tanda tangan akan dianggap palsu, dan apabila lebih besar atau sama dengan *threshold* akan dianggap sebagai asli. Pada penelitian ini digunakan 50 titik threshold dari threshold minimum sampai ke threshold maksimum.

Untuk melakukan pengukuran kinerja sistem verifikasi tanda tangan, digunakan AUC (Area Under Curve) dari kurva ROC (Receiver Operating Characteristic). Kurva ROC adalah kurva dua dimensi dimana TPR (True Positive Rate) / Recall diplot pada sumbu y dan FPR (False Positive Rate) diplot pada sumbu x pada titik threshold yang bervariasi. Pada penelitian ini digunakan 50 threshold dan kemudian dibangun kurva berdasarkan nilai TPR dan FPR pada masing-masing threshold. Sedangkan AUC adalah luas yang dibentuk dibawah kurva. Nilai AUC memiliki interpretasi yaitu nilai kemungkinan tanda tangan asli yang terverifikasi sebagai asli daripada tanda tangan palsu yang terverifikasi sebagai asli. Artinya apabila nilai AUC lebih besar maka lebih baik karena kemungkinan True Positive nya lebih besar. Nilai AUC berguna khususnya dalam studi perbandingan antara dua sistem diagnostik. Apabila dua sistem dibandingkan, lebih disarankan menggunakan keseluruhan ROC daripada hanya satu titik tertentu (Swets, 1979). Nilai maksimum dari AUC adalah 1, dengan kurva berbentuk segitiga dari ujung kiri (0,1) dan ujung kanan (1,1) hal tersebut terjadi jika tidak ada False Negative di setiap threshold. Sedangkan nilai AUC = 0.5 merupakan garis diagonal dari (0,0) dan (1,1) dan itu merupakan performa dari random verifikator yaitu verifikator yang hanya menebak tanpa data latih pada kasus klasifikasi biner. Apabila nilai AUC dibawah 0.5 itu bisa jadi kesalahan dalam algoritma verifikator atau fitur-fitur yang tidak relevan karena hasilnya yang lebih buruk dari random verifikator.

Pada penelitian kali ini, digunakan dua dataset yaitu dataset SVC2004 dan dataset Privat. Penggunaan Dynamic Time Warping sebagian besar menghasilkan kinerja yang lebih buruk daripada Manhattan Distance. Nilai AUC sistem verifikasi dengan metode DTW selalu lebih kecil daripada sistem verifikasi dengan metode Manhattan Distance. Pada dataset SVC2004 kasus pemalsuan terlatih nilai AUC DTW lebih buruk dengan selisih 0.02623. Pada kasus pemalsuan acak nilai AUC DTW juga lebih buruk dengan selisih 0.00029. Hal yang sama juga didapatkan

pada dataset Privat. Pada kasus pemalsuan terlatih nilai AUC DTW lebih kecil dengan selisih 0.053600166. Pada kasus pemalsuan acak nilai AUC DTW lebih kecil dengan selisih 0.047.

Sedangkan pada uji kedua, penggunaan fitur-fitur baru yang diturunkan dari histogram X dan Y memberikan kinerja yang lebih baik. Nilai AUC sistem dengan fitur-fitur baru selalu lebih baik atau lebih besar. Pada dataset SVC2004 kasus pemalsuan terlatih nilai AUC sistem dengan fitur baru lebih besar dengan selisih 0.00654. Sedangkan pada kasus pemalsuan acak nilai AUC sistem dengan fitur-fitur baru lebih besar dengan selisih 0.0549. Pada dataset Privat kasus pemalsuan terlatih nilai AUC sistem dengan fitur baru lebih besar dengan selisih 0.024. Juga pada kasus pemalsuan acak nilai AUC lebih besar dengan selisih 0.08. Fitur-fitur baru yang paling berpengaruh terhadap kinerja pada dataset SVC2004 adalah 12 fitur dengan urutan sebagai berikut $x_4, x_3, x_1, x_2, y_6, x_8, x_7, y_8, y_7, y_1, y_3, y_2$. Sedangkan fitur-fitur baru yang paling berpengaruh terhadap kinerja pada dataset Privat adalah 12 fitur dengan urutan sebagai berikut $x_6, x_5, x_3, x_4, y_5, y_4, x_7, y_7, y_6, y_8, x_2$, dan y_2 . Dua belas fitur-fitur baru tersebut juga memiliki peringkat yang tinggi dan konsisten dengan menggunakan metode pemeringkatan yang berbeda. Tinggi karena 12 fitur berada di bawah peringkat 214 atau setengah dari 428 dan konsisten karena selisih antar peringkat kecil.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan di atas, maka permasalahan utama pada penelitian dapat dirumuskan sebagai berikut:

- a Bagaimana kinerja dari sistem verifikasi tanda tangan online yang menggunakan DTW sebagai pengukur jarak antara tanda tangan uji dengan template apabila dibandingkan dengan *Manhattan Distance* sebagai pengukur jarak ?
- b Bagaimana kinerja dari sistem verifikasi tanda tangan online dengan penambahan fitur histogram X dan Y apabila dibandingkan dengan tanpa menggunakan fitur histogram X dan Y ?
- c Bagaimana peringkat fitur-fitur baru tersebut dibandingkan dengan fitur-fitur yang lain ?

1.3. Tujuan Penelitian

Berdasarkan latar belakang dan rumusan masalah yang telah dijelaskan sebelumnya, maka penelitian ini bertujuan:

- a Membandingkan kinerja sistem verifikasi yang menggunakan DTW sebagai penghitung jarak dibandingkan dengan sistem verifikasi yang menggunakan *Manhattan Distance* sebagai penghitung jarak
- b Membandingkan kinerja sistem verifikasi yang sudah ditambahkan fitur dari histogram X dan Y dibandingkan dengan sistem verifikasi tanpa menggunakan fitur dari histogram X dan Y .
- c Identifikasi peringkat fitur dalam kemampuannya melakukan verifikasi tanda tangan *online*.

1.4. Ruang Lingkup Penelitian

Penelitian ini memiliki beberapa batasan antara lain sebagai berikut:

- a Dataset publik tanda tangan dinamis yang digunakan untuk pengujian kinerja adalah SVC2004 dan dataset Privat dibuat sendiri.
- b Dataset privat tanda tangan diambil dengan menggunakan web browser dengan memanfaatkan teknologi HTML 5 dan canvas.
- c Perangkat yang digunakan dalam pengambilan data tanda tangan adalah perangkat *mobile* dan diambil dengan menggunakan jari.

1.5. Kontribusi Penelitian

Penelitian ini memiliki kontribusi dalam lingkup teori dan juga dalam lingkup praktis. Dalam lingkup teori, penelitian ini mencoba menambahkan fitur-fitur yang diturunkan dari histogram X dan Y yang tidak digunakan dalam penelitian sebelumnya. Dengan begitu, diharapkan penambahan tersebut mampu meningkatkan kinerja yang dapat dilihat dari nilai AUC yang lebih besar. Selain itu, dalam penelitian ini juga mencoba algoritma *Dynamic Time Warping* yang disarankan pada penelitian sebelumnya, untuk mengukur jarak antara tanda tangan uji dengan template yang sudah disimpan. Pada penelitian sebelumnya, jarak tersebut dihitung dengan algoritma *Manhattan Distance*. DTW dikenal oleh banyak peneliti di bidang verifikasi tanda tangan *online* karena kemampuannya dalam

menghitung tingkat ketidaksamaan suatu vektor. Yang terakhir, penelitian ini juga memperkaya khasanah tentang verifikasi tanda tangan online/dinamis.

Dalam lingkup praktis, metode yang diusulkan diharapkan bisa digunakan sebagai metode baru dalam proses penanda tangan, sehingga proses tanda tangan dokumen dapat dilakukan secara elektronik. Dengan begitu diharapkan proses persetujuan dokumen dalam perkantoran akan lebih cepat. Verifikasi tanda tangan dinamis juga dapat digunakan untuk mendukung sistem notadinas elektronik dan mendukung konsep *papperless office* untuk mengurangi penggunaan kertas.

1.6. Sistematika Penulisan Dokumen

Sistematika penulisan dokumen laporan penelitian tesis ini dibagi menjadi lima bab, yaitu sebagai berikut:

BAB 1 PENDAHULUAN

Pada bab ini dijelaskan mengenai latar belakang, rumusan masalah, tujuan penelitian, ruang lingkup penelitian, kontribusi penelitian dan sistematika penulisan.

BAB 2 LANDASAN TEORI DAN KAJIAN PUSTAKA

Pada bab ini dijelaskan mengenai kajian pustaka dari berbagai penelitian yang berkaitan dengan penelitian ini. Kajian pustaka ini bertujuan untuk memperkuat dasar dan alasan dilakukannya penelitian ini. Selain kajian pustaka, pada bab ini juga dijelaskan mengenai teori-teori terkait yang bersumber dari buku, jurnal, maupun artikel yang berfungsi sebagai dasar dalam melakukan penelitian agar dapat memahami konsep atau teori penyelesaian permasalahan yang ada. Teori yang dijelaskan antara lain mengenai Kedinamisan Tanda Tangan, Fitur Tanda Tangan Dinamis, Fitur Parameter dari Metode Statistik, Histogram, Dataset Tanda Tangan, Template Tanda Tangan, Dynamic Time Warping, Kategori Pencocokan Fitur, dan Pengukuran Performa, Pemingkatan Fitur Berbasis Fitur.

BAB 3 METODOLOGI PENELITIAN

Pada bab ini dijelaskan mengenai langkah–langkah penelitian beserta metode yang digunakan. Langkah–langkah penelitian dijelaskan dalam sebuah diagram alur yang sistematis dan akan dijelaskan tahap demi tahap.

BAB 4 UJI COBA DAN ANALISIS HASIL

Pada bab ini akan dijelaskan uji coba terhadap tahap pemilihan fitur berdasarkan skenario uji coba yang telah dirancang sebelumnya. Selain itu pada bab ini juga dijelaskan mengenai analisis hasil uji coba.

BAB 5 KESIMPULAN DAN SARAN

Pada bab ini berisi kesimpulan dari penelitian dan juga saran bagi penelitian berikutnya yang berasal dari kekurangan ataupun temuan dari penelitian ini.

(lembar ini sengaja dikosongkan)

BAB 2

LANDASAN TEORI DAN KAJIAN PUSTAKA

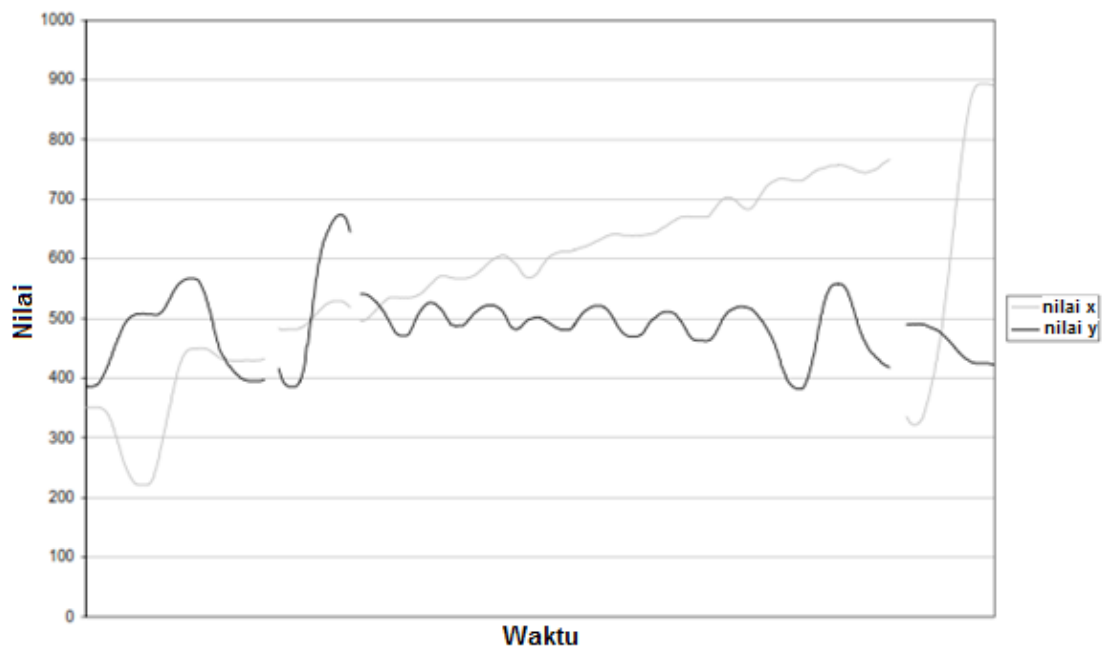
Pada bab ini dijelaskan mengenai teori-teori yang mendasari penelitian dan kajian pustaka mengenai penelitian–penelitian yang terkait. Teori yang dijelaskan antara lain mengenai Kedinamisan Tanda Tangan, Histogram, Dataset Tanda Tangan, Template Tanda Tangan, Dynamic Time Warping, Kategori Pencocokan Fitur, dan Pengukuran Performa.

2.1. Kedinamisan Tanda Tangan

Kedinamisan tanda tangan yang dapat ditangkap oleh tablet grafis diberikan oleh:

$$S(t) = [x(t), y(t), p(t)]^T \quad t = 0, 1, 2, \dots, n \quad (2-1)$$

Persamaan tersebut merupakan vektor kumpulan nilai lokasi x,y dari pen dan nilai tekanan dari pen pada interval waktu yang sama. Beberapa perangkat juga menangkap azimuth dan elevasi. Banyak tablet memiliki sampling rate 200 kali per sekon dan resolusi dari perangkat tersebut biasanya 1000 pixels/inch walaupun beberapa memiliki resolusi yang lebih baik. Tanda tangan orang amerika pada umumnya menggunakan penulisan nama orang dan oleh karena itu untuk tanda tangan orang amerika nilai x secara umum tumbuh secara linear seiring dengan waktu dengan sedikit okulasi pada kurva linear sedangkan nilai y menunjukkan variasi okulasi yang lebih banyak terhadap waktu, sering menjadi positif dan negatif selama tanda tangan. Sebuah contoh ditunjukkan dalam gambar 2.1.



Gambar 2.1. Grafik x dan y dari tanda tangan

Data tanda tangan, dapat digunakan untuk menghitung turunan dari x, y. Turunan pertama dari x dan y adalah kecepatan dalam 2 arah (yang dapat dikombinasikan untuk menghitung kecepatan total, jika diperlukan) dan turunan kedua adalah akselerasi dalam arah x dan y. Ada juga yang mungkin menghitung turunan ketiga walaupun ini tidak umum digunakan dalam sistem verifikasi tanda tangan. Kecepatan perubahan akselerasi biasa disebut dengan *jerk*. Setelah turunan-turunan ini sudah dihitung, (berasumsi bahwa tidak ada turunan dari pressure/tekanan) data berikut ini akan tersedia :

$$S(t) = [x(t), y(t), p(t), x_v(t), y_v(t), x_a(t), y_a(t), x_j(t), y_j(t)]^T \quad t = 0, 1, 2, \dots, n \quad (2-2)$$

Sudah jelas bahwa setiap kali seorang melakukan tanda tangan, jumlah contoh/sampel yang diperoleh akan berbeda. Variasi dalam tanda tangan dari orang

yang sama ini akan membuat sulit untuk membandingkan satu set nilai fitur dari tanda tangan yang satu dengan satu set nilai tanda tangan yang lain.

2.1.1. Kecepatan dan Akselerasi Tanda Tangan

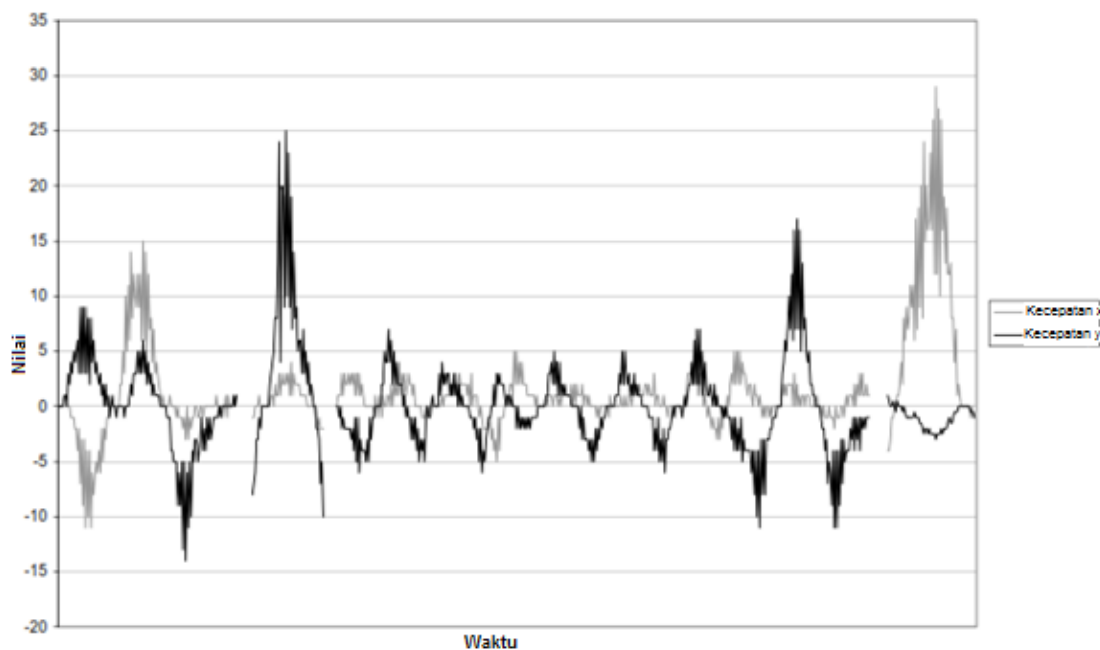
Tanda tangan dapat diartikan sebagai deretan dari goresan/stroke/garis. (Dimauro, Impedovo, & Pirlo, 1994) mendefinisikan garis/stroke sebagai deretan dari komponen-komponen fundamental, dibatasi oleh pemotongan secara curam/terjal dan tiba-tiba. Mari berasumsi, misalkan garis dalam arah y dari nilai y kecil ke nilai y yang besar dan selama periode tersebut abaikan perpindahan apapun dalam arah x . Jika kecepatan dalam arah y selama proses terbentuknya garis diteliti, akan ditemukan bahwa garis dapat dikarakterkan dengan tumbuhnya nilai kecepatan yang akan sampai ke puncak di pertengahan garis dan kemudian diikuti dengan turunya nilai kecepatan. Ini akan terlihat jelas jika kita melihat grafik posisi garis- y yang seperti huruf "S" dalam gambar 2 dan grafik kecepatan y yang berkaitan dalam gambar 4. Grafik kecepatan berbentuk gelombang, kecepatan berawal dari 0, mencapai puncak dan berakhir dengan kecepatan 0. Selain itu, jika kita mempelajari grafik percepatan saat kecepatan sampai di puncak, ditemukan bahwa percepatan tumbuh dari nilai 0 sampai mencapai nilai puncak di titik pertengahan saat grafik kecepatan sedang naik ke puncak dan kemudian berkurang secara stabil sampai ke 0 pada saat kecepatan mencapai puncak.

Secara teori grafik kecepatan seharusnya memiliki puncak dan lembah (extrema) ganda dibanding dengan grafik x dan y dan grafik akselerasi akan memiliki 4 kali jumlahnya. Namun, seperti yang kita lihat pada praktek, kondisinya tidak selalu demikian disebabkan oleh jitter. Selain itu, tinggi extrema dari grafik kecepatan akan lebih kecil daripada panjang garis dalam tanda tangan dan tinggi extrema dari grafik percepatan akan lebih kecil lagi. Ini digambarkan secara jelas oleh grafik kecepatan dan grafik percepatan dalam figur 4 dan figur 6 yang menunjukkan bahwa mayoritas nilai kecepatan antara -5 dan 5 sementara sebagian besar nilai akselerasi antara -1 dan 1.

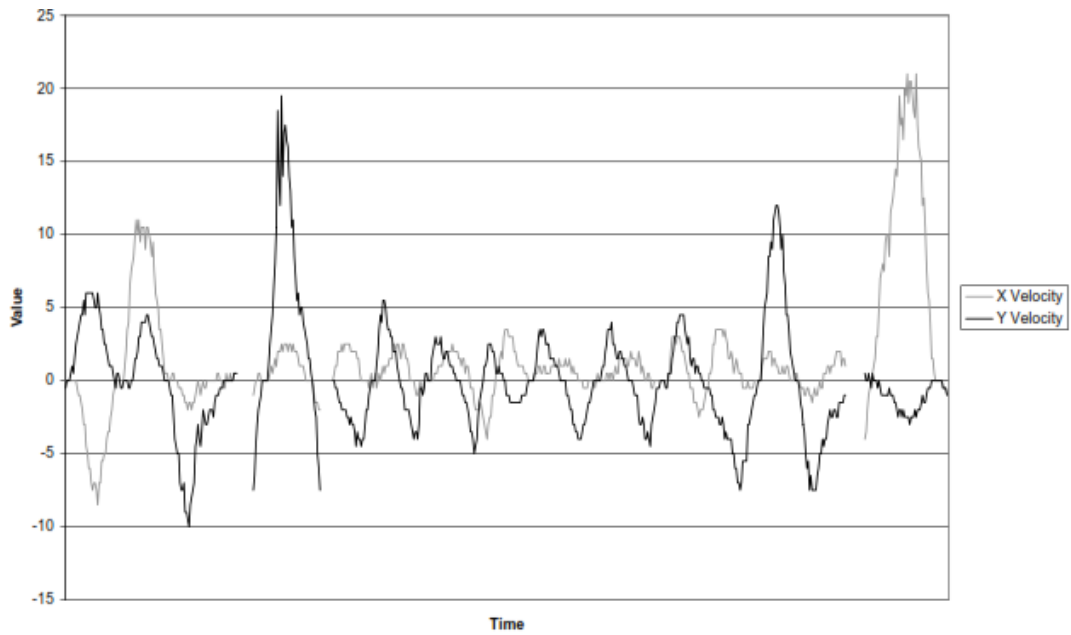
Bentuk grafik kecepatan garis telah dipelajari oleh (Plamondon, 1993) yang menyatakan bahwa bentuk grafik kecepatan yang didapatkan saat garis dengan pergerakan cepat (seperti pada tanda tangan), diperkirakan berbentuk seperti lonceng

tetapi bentuknya tidak simetris dan diklaim bahwa bentuk ini akan konsisten untuk pergerakan dengan durasi, jarak, dan kecepatan puncak yang berbeda-beda. Bentuk yang konsisten ini bisa jadi berkaitan dengan cara sistem saraf pusat berencana dan mengontrol pergerakan.

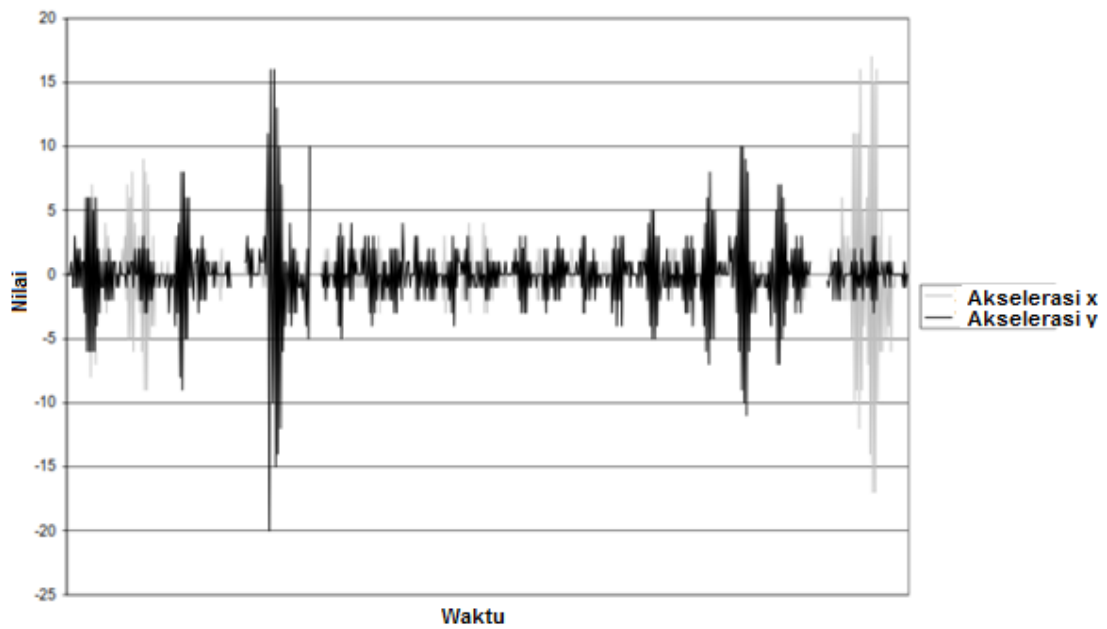
Mengingat bahwa, nilai posisi hanya akurat sampai satu piksel, perkiraan nilai kecepatan dan akselerasi dapat memiliki kesalahan yang signifikan. Sebagai contoh, Profil kecepatan pada gambar 3 didapatkan dengan menggunakan metode perbedaan penerusan sederhana (*simple forward difference*) sementara profil kecepatan pada gambar 4 didapatkan dengan perbedaan terpusat (*central difference*). Banyak jitter dalam gambar 3 dieliminasi dengan menggunakan formula yang lebih akurat dalam gambar 4. Perbedaan bahkan lebih jelas ketika membandingkan dua grafik akselerasi dalam gambar 5 dan gambar 6. Perbedaan yang besar dalam nilai maksimum dalam gambar 5 dan gambar 6 harus dicatat. Jitter masih cukup signifikan dalam nilai akselerasi dalam gambar 6.



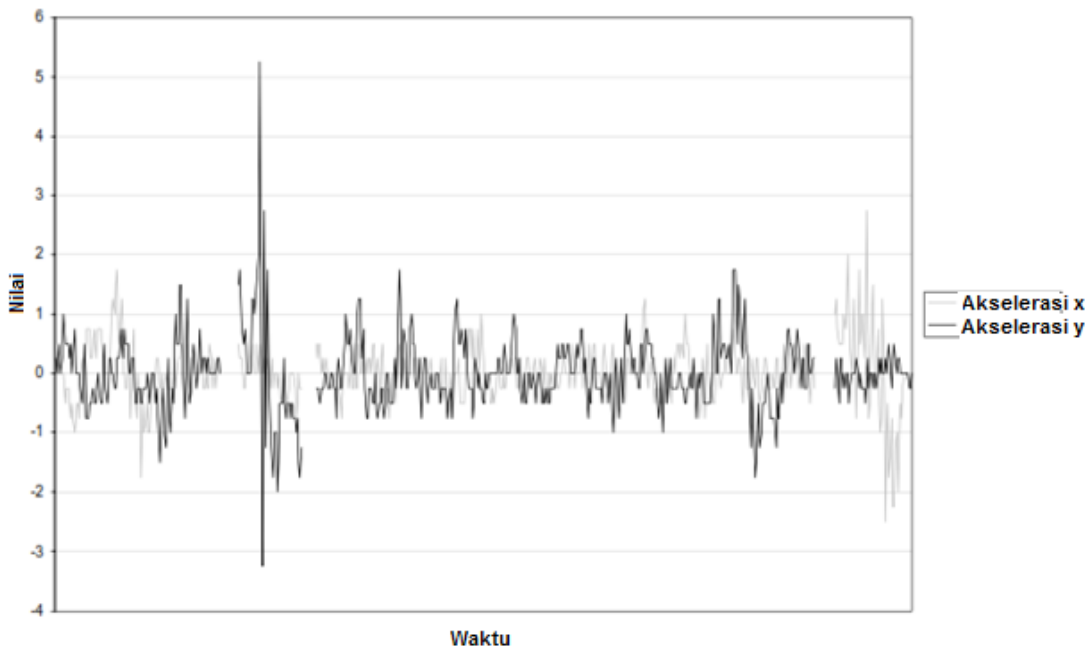
Gambar 2.2. Kecepatan x dan y dari tanda tangan menggunakan *Forward Differences*



Gambar 2.3. Kecepatan x dan y dari tanda tangan menggunakan Central Differences



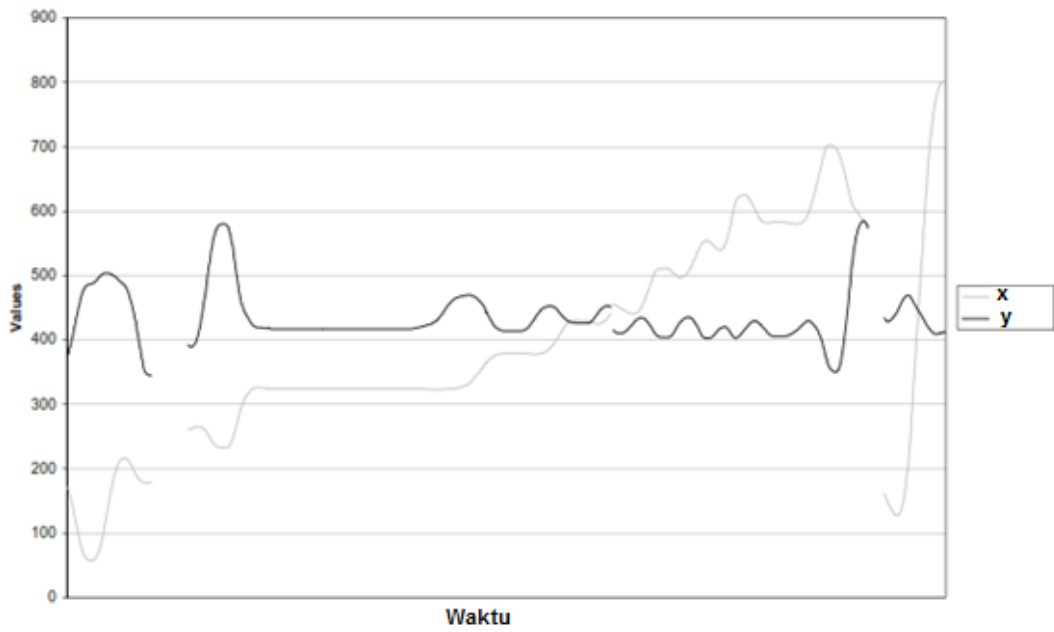
Gambar 2.4. Percepatan x dan y menggunakan Forward Differences



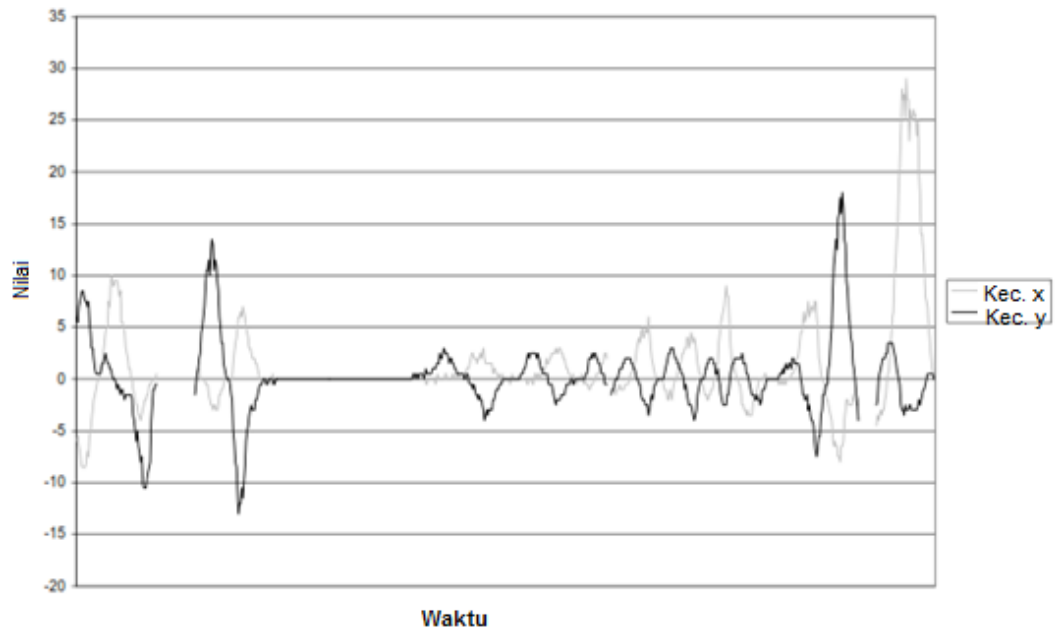
Gambar 2.5. Percepatan x dan y menggunakan Central Differences

2.1.2. Contoh Pemalsuan

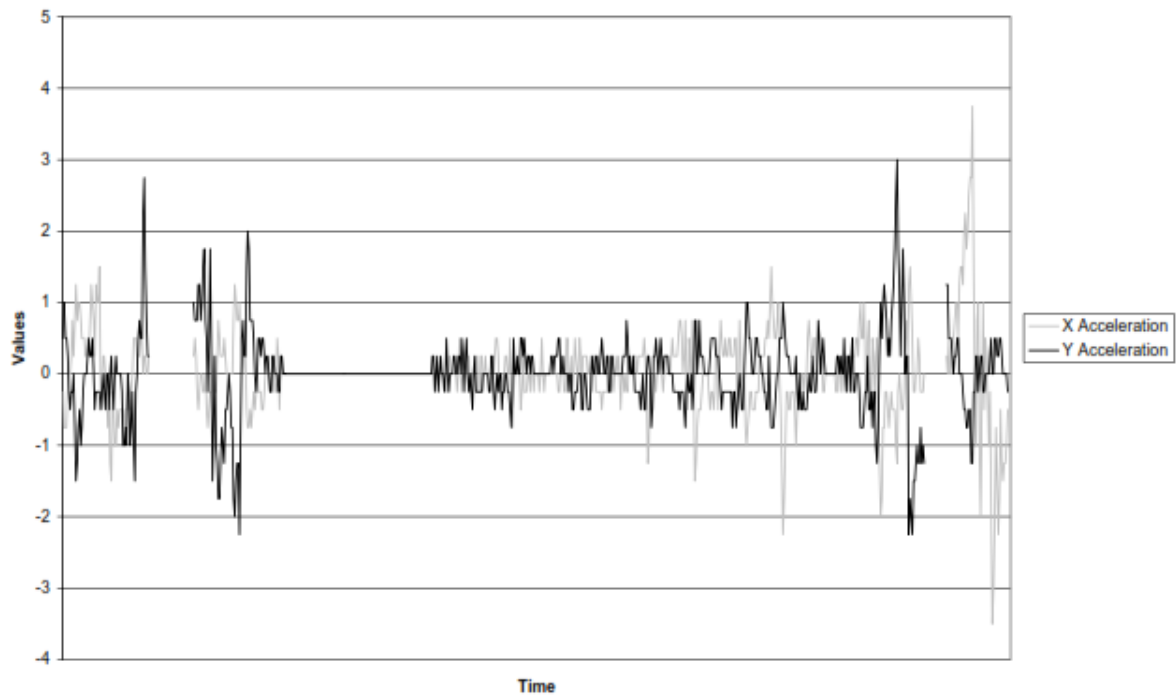
Sebelumnya sudah dibahas tentang kedinamisan dari tanda tangan, sekarang bagaimana pemalsu memalsukan sebuah tanda tangan dan seberapa sulit bagi pemalsu untuk menghasilkan pemalsuan yang bagus? Seperti yang dicatat sebelumnya bahwa seseorang tidak bisa menulis tanda tangan orang lain dalam pergerakan yang cepat tanpa latihan yang banyak sebelumnya dan oleh karena itu untuk menghasilkan pemalsuan yang bagus tidaklah mudah. Gambar 7, gambar 8, dan gambar 9 menunjukkan grafik posisi, kecepatan, dan akselerasi yang menggambarkan pemalsuan berskill (*skilled forgeries*) dari tanda tangan dalam gambar 2. Gambar-gambar ini menunjukkan bahwa grafik pemalsuan ini sangat berbeda dengan grafik asli dari tanda tangan dalam gambar 2, 4, dan 6. Perbedaan yang paling terlihat adalah waktu dari “pen-up”. Dan juga, sebagian besar nilai kecepatan dan percepatan pemalsuan dalam kasus ini lebih kecil daripada tanda tangan yang asli.



Gambar 2.6. x dan y dari tanda tangan palsu



Gambar 2.7. Kecepatan x dan y tanda tangan palsu menggunakan Central Differences



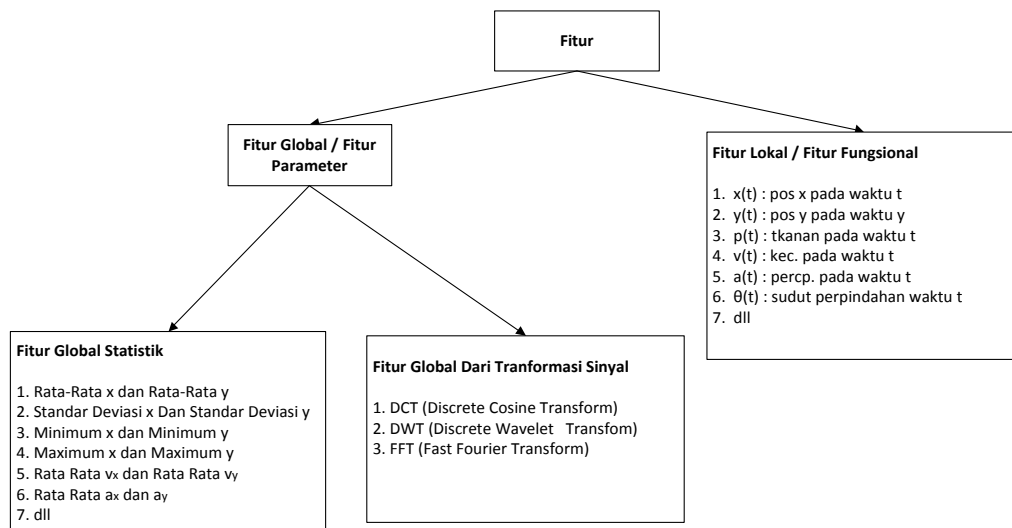
Gambar 2.8. Percepatan x dan y tanda tangan palsu menggunakan Central Differences

Seperti yang dicatat sebelumnya, beberapa tanda tangan lebih mudah dipalsukan daripada yang lainnya. Kompleksitas pemalsuan tanda tangan dipelajari oleh (J.-J. Brault & Plamondon, 1989; J. J. Brault & Plamondon, 1993) dalam percobaannya memperkirakan resiko intrinsik sebuah tanda tangan dapat dipalsukan. Mereka mencatat bahwa manusia hanya dapat mengingat 7 variasi pola tanpa kesalahan, jadi pemalsu tidak dapat mengingat semua detail tentang tanda tangan yang sedang dipalsukan. Proses meminimalkan ingatan dengan melakukan perekaman informasi yang perlu untuk diingat, harus dilakukan. Sebagai contoh, pemalsu akan merekam informasi tanda tangan John Smith, pertama dengan mengingat namanya dan kemudian mengingat bahwa tanda tangan John Smith mempunyai J yang lebih panjang daripada yang biasa ditulis secara normal oleh pemalsu dan memiliki huruf S yang lebih bulat dan lain sebagainya. Berdasarkan observasi ini, mereka membuat formula/ekspresi untuk menghitung koefisien tingkat kesulitan pemalsuan tanda tangan. Intinya, formula ini mendefinisikan

tingkat kesulitan tanda tangan adalah dengan fungsi variation rate dalam panjang dan arah dari garis/lintasan.

2.2. Fitur Tanda Tangan Dinamis

Proses pemilihan fitur berdampak sangat signifikan terhadap performansi verifikasi tanda tangan dinamis atau online. Oleh sebab itu, ekstraksi fitur dikategorikan sebagai salah satu langkah yang paling penting dalam sistem verifikasi tanda tangan (GURRALA, 2011; Hao, 2011; Taherzadeh, Karimi, Ghobadi, & Beh, 2011). Fitur yang diseleksi harus bersifat diskriminatif yaitu bisa dibedakan antara tanda tangan asli dan tanda tangan palsu. Ada dua jenis tipe fitur yang bisa memvalidasi tanda tangan. Pertama disebut dengan fitur global. Kedua disebut dengan fitur lokal (Corradin, 2008; S Garcia-Salicetti et al., 2009; Taherzadeh et al., 2011).



Gambar 2.9. Kategori Fitur Tanda Tangan

Sistem verifikasi tanda tangan yang menggunakan fitur global disebut dengan pendekatan parameter. Sedangkan verifikasi tanda tangan yang menggunakan fitur lokal disebut dengan pendekatan fungsional.

Sebuah fitur disebut fitur global saat dia diambil/diekstrak dari tanda tangan secara keseluruhan. Ini berarti bahwa nilai-nilai x dan nilai-nilai y perlu disusutkan kedalam satu nilai (Al-Omari, Abdullah, & Omar, 2011; Corradin, 2008; S Garcia-

Salicetti et al., 2009). Contoh fitur global adalah *density* dan *wavelet transform*, rata-rata nilai x dan y , rata-rata tekanan, kemiringan pen, rata-rata kecepatan, korelasi koordinat, durasi tanda tangan, standard deviasi x dan y .

Sedangkan dalam pendekatan fungsional keseluruhan sinyal dalam satuan waktu $(x(t), y(t), p(t))$ akan ditambahkan kedalam kumpulan fitur, dan perbandingan akan dilakukan dari titik ke titik antara tanda tangan referensi/template dengan randa tangan uji (Feng & Wah, 2003). Oleh karena itu, fitur yang diambil dengan pendekatan fungsional adalah fitur lokal. Fitur lokal adalah fitur yang berhubungan dengan titik sampel secara spesifik dalam lintasan/garis tanda tangan (Tahezadeh et al., 2011). Selain posisi x dan y seperti yang disebutkan diatas, turunan pertama dan kedua dari posisi x dan y seperti kecepatan x dan y serta percepatan x dan y juga termasuk kedalam fitur lokal.

2.3. Fitur Parameter dari Metode Statistik



Gambar 2.10. Pengambilan Fitur Parameter Statistik

Secara umum, fitur global diambil dengan proses seperti diatas. Data diambil dari tablet/perangkat akuisisi pada saat proses tanda tangan yaitu $x(t), y(t), p(t)$. Dari data tersebut dapat diturunkan menjadi data baru seperti kecepatan, percepatan, dan jerk. Setelah mendapatkan semua data yang diperlukan, maka dilakukan penghitungan misalnsya rata-rata, minimum dan maksimum, untuk mengubah kumpulan dari fitur lokal tersebut menjadi fitur global. Dengan memperhatikan jumlah fitur, sering ada kecenderungan untuk melibatkan fitur yang lebih banyak dengan harapan akan meningkatkan performansi. Namun menggunakan jumlah fitur yang banyak tidak selalu menghasilkan performansi yang lebih baik dan bisa menambah masalah baru. Sebagai contoh, jika sebuah metode menggunakan terlalu banyak fitur, tempat penyimpanan / storage yang dibutuhkan untuk menyimpan seluruh fitur tersebut akan relatif besar. Tidak ada

dua tanda tangan asli, memiliki fitur yang identik, jadi sebuah tanda tangan asli tidak akan memiliki semua nilai fitur yang sama dengan tanda tangan referensi.

Beberapa studi telah meneliti tentang fitur-fitur global. Fitur-fitur global dengan jumlah yang besar dipelajari oleh (Crane & Ostrem, 1983) dengan menggunakan pen yang sudah diinstrumentasi untuk mengambil tiga gaya saat penulisan (gaya arah bawah, gaya x dan gaya y) dan kemudian dihitung 44 fitur global, beberapa dari mereka ditunjukkan dalam tabel 1. Metode yang digunakan untuk memilih fitur adalah dengan menghilangkan satu fitur secara bergantian dari kumpulan fitur global dan menemukan fitur yang dengan penghapusannya memberikan ERR yang paling rendah. Metode terus berlanjut sampai penghapusan sebuah fitur tidak mengurangi ERR. Selanjutnya 50 fitur dipilih. Untuk mengevaluasi metode yang diproposalkan, sebuah database terdiri dari 5220 tanda tangan asli dari 58 orang dikumpulkan selama 4 bulan. Selain itu, database juga terdiri dari 648 tanda tangan palsu dengan kategori pemalsuan ber-skill dari 12 orang pemalsu. ERR dengan angka 1.5% dicapai walaupun setengah dari tanda tangan asli digunakan untuk data latih untuk memilih fitur-fitur yang digunakan sementara setengah yang lain digunakan untuk data uji.

(Lee, Berger, & Aviczer, 1996) menggunakan 42 fitur, diantara fitur-fitur tersebut terdaftar dalam tabel 1. Sejumlah teknik untuk pemilihan fitur digunakan. Salah satu teknik yaitu untuk setiap fitur, rata-rata untuk subjek i dibandingkan dengan rata-rata dari fitur yang sama dari semua subjek. Nilai maksimum dari perbandingan jarak dihitung untuk setiap subjek dan setiap fitur. Tingkat kepentingan setiap fitur dari individu diberikan oleh jarak maksimum ini. Algoritma ini digunakan untuk menemukan subset fitur dari setiap individu. Ditunjukkan bahwa 23 atau 24 fitur memiliki performansi yang paling baik. 10 dari mereka ada di tabel 1.

(Ketabdar, Richiardi, & Drygajlo, 2005) memberikan sebuah metodologi untuk menyeleksi fitur global yang paling diskriminatif. Lebih dari 150 fitur global dari 60 paper dipelajari dan subset pertama yang diambil terdiri dari 46 fitur untuk diinvestigasi. Untuk melakukan seleksi fitur, algoritma *near optimal feature space search* digunakan untuk menghindari *exhaustive search*. Fungsi berbasis biaya dalam kelas dengan keanekaragaman kecil dan antar kelas dengan keanekaragaman

besar digunakan. Dengan tujuan untuk memperhitungkan dampak korelasi antar komponen vektor, cost/biaya dihitung pada keseluruhan vektor daripada satu per satu. Hasil dari penggunaan metode terhadap subset 50 pengguna (50 asli dan 50 palsu) dari database MCYT menghasilkan 12 fitur yang terdaftar di tabel 1.

(Fierrez-Aguilar, Nanni, et al., 2005) mengevaluasi 100 fitur global untuk menemukan nilai diskriminatif mereka. Ini dilakukan dengan menghitung jarak Mahalonobis antara rata-rata dari tanda tangan latih dari satu orang tertentu dengan kumpulan seluruh data latih dari semua orang. Fitur-fitur kemudian diurutkan berdasarkan class separability inter-user. 10 fitur yang paling diskriminatif terdaftar dalam tabel 1.

Sejumlah peneliti yang lain juga telah mempelajari fitur-fitur global. Sebagai contoh (Richiardi, Ketabdar, & Drygajlo, 2005), mempelajari 46 fitur global dan menggunakan database MYCT untuk memilih 12 fitur global terbaik. Diantaranya adalah Average Velocity, Average Pressure, Total Time, pen down samples, average and minimum pressure. Daftar fitur-fitur global dapat dilihat dalam tabel 1. Kesimpulan menunjukkan bahwa fitur global terbaik adalah total waktu dan rata-rata kecepatan

Tabel 2.1. Penelitian Terkait Fitur Global Statistik

Referensi	Fitur Yang Dipelajari	Fitur Terbaik	Metode Pencocokan
(Crane dan Obstern 1983) X adalah gaya arah x Y adalah gaya arah y P adalah gaya arah bawah	Setiap X, Y dan P: 1. Scaled Mean 2. Standard Deviation 3. Minimum 4. Maximum 5. Average Absolute 6. Average Posistive 7. Number Of Positive Samples 8. Average Negative 9. Number of Negative Samples 10. Number of 0 crossing 11. Maximum Minus Scaled Mean 12. Maximus Minus Minimum 13. Scaled Mean Minus Minimum Tidak tergantung X Y dan P : 14. Total Time 15. Number of Segments -1 16. Time Up 17. Number Of Segments 18. Time Down	1. Total Time 2. Time Up 3. Time Down 4. Scaled Mean X and Y 5. Standard Deviation X and Y and P 6. Minimum X and Y 7. Mean Positive X and Y 8. Max Y 9. Mean Absolute P 10. Mean Posotive P 11. Numbers of Negative P samples 12. Number Of 0 crossing Y and P 13. Maximum Minus Scaled Mean Y and P 14. Scaled Mean Minus Minimum Y	Euclidean Distance

(Lee et.al. 1996)	<ol style="list-style-type: none"> 1. Average Writing Speed 2. Maximum Writing Speed 3. Time of Max Speed 4. $t(1^{st} \text{ pen move}) - t(1^{st} \text{ pen down})$ 5. Total Signing Duration 6. Total Pen Down Duration 7. Minimum horizontal writing speed 8. Time of Feature 7 9. Total dots recorded 10. Average dot execution time 11. Number pen ups 12. Time of 2nd pen down 13. Initial direction 14. Direction from 1st to 2nd pen down 15. dll 		Majority Classifier
-------------------	--	--	---------------------

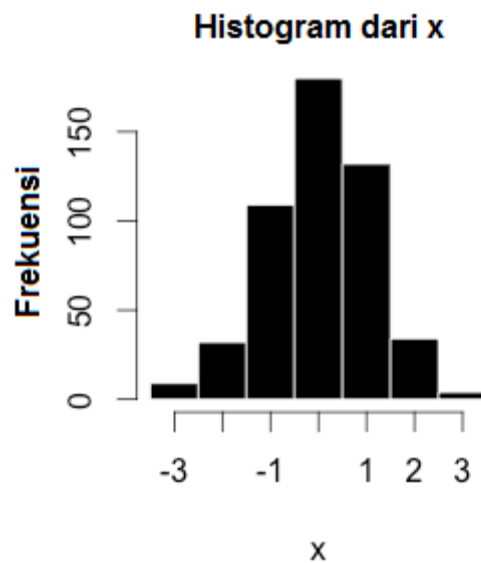
2.4. Histogram

Histogram adalah sebuah representasi grafik yang akurat tentang distribusi data numerik. Histogram merupakan pendekatan atau perkiraan dari distribusi probabilitas dari *continuous variable* (variabel kuantitatif) dan pertama kali dikenalkan oleh Karl Pearson. Itu merupakan jenis dari graph bar. Untuk membangun histogram, langkah pertama adalah membuat “bin” dari jarak nilai yaitu membagi seluruh jarak nilai ke dalam deretan interval dan kemudian menghitung berapa banyak nilai yang jatuh kedalam setiap interval. Bin biasanya dibuat berurutan yaitu interval yang tidak tumpang tindih. Bin harus berdekatan dan sering (tapi tidak harus) memiliki ukuran interval yang sama.

Apabila bins berukuran sama, sebuah kotak akan diarahkan diatas bin dengan tinggi yang proporsional terhadap frekuensi, yaitu jumlah nilai yang masuk ke dalam bin. Sebuah histogram dapat juga dinormalisasi untuk menampilkan frekuensi relatif. Kemudian histogram akan menampilkan proporsi nilai yang masuk kedalam setiap kategori, sehingga jumlah tinggi semua bin adalah sama dengan satu. Jadi, ada dua jenis frekuensi dalam histogram. Yang pertama adalah frekuensi absolut yaitu jumlah frekuensi atau penghitungan yang asli dari elemen yang masuk kedalam bin tertentu. Yang kedua adalah frekuensi relatif, yaitu frekuensi absolut yang telah dinormalisasi oleh jumlah total elemen dalam histogram.

Akan tetapi, bins tidak harus memiliki lebar yang sama. Dalam kasus tersebut kotak dibuat dengan luas yang proporsional terhadap frekuensi. Sedangkan tinggi kotak bukan menunjukkan frekuensi namun menunjukkan densitas frekuensi

yaitu jumlah nilai per unit dari lebar kotak. Oleh karena satu bin dengan bin lain yang berdekatan tidak ada jarak, maka kotak dari histogram saling bersentuhan satu sama lain untuk mengindikasikan bahwa variabel yang asli bersifat *continuous*. Histogram memberikan gambaran yang kasar tentang densitas dari distribusi data, dan sering untuk memperkirakan *probability density functions* dari variabel yang bersangkutan. Histogram bisa menggambarkan distribusi dari satu atribut. Histogram seperti itu, disebut dengan histogram 1 dimensi. Selain itu, histogram bisa juga menangkap distribusi dari dua atribut secara bersamaan. Histogram seperti itu, disebut dengan histogram 2 dimensi. Histogram dua dimensi mampu menggambarkan distribusi korelasi antara satu atribut dengan atribut yang lain.



Gambar 2.11. Contoh Histogram 1 Dimensi

2.4.1. Lebar dan Jumlah bin

Ukuran bin yang berbeda dapat mengungkapkan fitur yang berbeda dari data. Menggunakan bin yang lebih lebar dimana densitas data didalamnya rendah, mengurangi *noise* karena keacakan sampling. Menggunakan bin yang lebih sempit dimana densitas data tinggi akan memberikan presisi terhadap estimasi densitas. Beberapa ilmuwan telah mencoba untuk menentukan jumlah bin yang optimal, akan tetapi metode-metode tersebut secara umum membuat asumsi-asumsi yang kuat

tentang bentuk distribusi. Bergantung dari distribusi dari data aktual dan tujuan analisis, lebar bin yang berbeda bisa jadi tepat, jadi eksperimen biasanya dibutuhkan untuk menentukan lebar bin yang tepat. Akan tetapi ada bermacam petunjuk dan aturan praktis yang berguna.

2.4.2. Fitur Histogram

Dalam penelitian ini, sebuah tanda tangan *online*, direpresentasikan oleh satu set histogram. Fitur-fitur histogram didesain untuk menangkap atribut-atribut yang penting dari tanda tangan dan juga relasi antara atribut-atribut ini. Ini perlu dicatat bahwa histogram digunakan secara luas sebagai fitur untuk menangkap atribut statistik dalam masalah rekognisi. Sebagai contoh, fitur histogram digunakan dalam permasalahan pengenalan objek (Schiele & Crowley, 1996) dan pengenalan tanda tangan *offline* (Qiao et al., 2007). Penggunaan histogram untuk verifikasi tanda tangan online pertama kali disarankan oleh (Nelson et al., 1994). Histogram juga telah digunakan sebagai bagian dari fitur set dalam (Fallah et al., 2011) dan (Fierrez-Aguilar, Nanni, et al., 2005). Akan tetapi, dalam (Fallah et al., 2011) dan (Fierrez-Aguilar, Nanni, et al., 2005), penggunaan histogram hanya terbatas pada sudut yang diturunkan dari vektor yang menghubungkan dua titik yang berurutan dalam tanda tangan *online*. Sebenarnya, lebih banyak informasi yang tersedia yang dapat digunakan untuk menghasilkan histogram yang berguna dalam verifikasi tanda tangan *online*. Semua itu termasuk trayektori x-y, kecepatan, sudut, tekanan, dan turunan mereka.

Proses ekstraksi fitur dimulai dengan mengkonversi data *time-series* dari tanda tangan kedalam deret vektor kartesian dan juga turunannya. Kemudian, setiap vektor kartesian juga dikonversi kedalam vektor dalam sistem koordinat polar. Akhirnya, histogram dari deret vektor-vektor ini dibuat. Detail dari proses ekstraksi fitur adalah sebagai berikut. Misalkan $X = \{x_1, x_2, \dots, x_n\}$, $Y = \{y_1, y_2, \dots, y_n\}$, dan $P = \{p_1, p_2, \dots, p_n\}$ adalah koordinat x, koordinat y, dan tekanan dari tanda tangan dengan panjang n yang disampling selama $T = \{t_1, t_2, \dots, t_n\}$. Sebagai catatan bahwa jika waktu interval tidak konstan, maka proses normalisasi terhadap X , Y , dan P perlu dilakukan dengan menggunakan informasi T , sebelum diproses oleh sistem. Untuk

dataset SVC2004, semua tanda tangan disampling dengan kecepatan yang sama. Oleh karena itu, tidak memerlukan normalisasi.

Pertama – tama vektor X^l , Y^l , dan P^l dan juga turunanya dihitung dengan persamaan sebagai berikut :

$$X^1 = \{x_i^1 | x_i^1 = x_{i+1} - x_i\}, \quad (2-3)$$

$$Y^1 = \{y_i^1 | y_i^1 = y_{i+1} - y_i\}, \quad (2-4)$$

$$P^1 = \{p_i\}, \quad (2-5)$$

dan.,

$$X^k = \{x_i^k | x_i^k = x_{i+1}^{k-1} - x_i^{k-1}\}, \quad (2-6)$$

$$Y^k = \{y_i^k | y_i^k = y_{i+1}^{k-1} - y_i^{k-1}\}, \quad (2-7)$$

$$P^k = \{p_i^k | p_i^k = p_{i+1}^{k-1} - p_i^{k-1}\}, \quad (2-8)$$

Dimana $k > 1$ dan $i = 1, 2, \dots, n-k$

Dengan menghitung selisih antara dua pasang titik yang saling berurutan seperti di atas, vektor X^l dan Y^l dapat menangkap fitur posisi yang bersifat invarian atau konsisten dari tanda tangan. Dan dengan mengulangi proses pengurangan ini sebanyak k kali maka akan menghasilkan turunan tingkat ke- k (k^{th}) dari X dan Y yang orisinal.

Kemudian, deret vektor $= \{v_i^* | i = 1, 2, 3 \dots n\}$, dibentuk dimana setiap elemen vektor, $v_i^* = \{v_i^* | v_i^1 \parallel \dots \parallel v_i^j\}$ adalah penggabungan dari v_i^k . Dimana v_i^k adalah 5 tupel yang terdiri dari turunan tingkat – k dari tekanan, koordinat kartesian, dan koordinat polar:

$$v_i^k = (x_i^k, y_i^k, r_i^k, \theta_i^k, p_i^k) \quad (2-9)$$

dimana

$$\theta_i^k = \tan^{-1} \left(\frac{y_i^k}{x_i^k} \right) \quad (2-10)$$

$$r_i^k = \sqrt{(x_i^k)^2 + (y_i^k)^2} \quad (2-11)$$

2.5. Dataset Tanda Tangan Publik

Dalam penelitian ini selain menggunakan dataset yang dibuat sendiri, juga digunakan dataset tanda tangan dinamis yang tersedia di Internet atau publik. Berikut adalah penjabaran detail mengenai pembuatan dataset tersebut.

2.3.1 Dataset SVC2004

SVC2004 (Yeung, Chang, Xiong, & George, 2004) adalah kompetisi tingkat internasional yang pertama dalam bidang verifikasi tanda tangan. Kompetisi ini terdiri dari dua *task* yang berbeda, dimana setiap *task* berdasarkan dataset tanda tangan yang berbeda. Data tanda tangan untuk *task* yang pertama hanya berisi data koordinat sedangkan untuk *task* kedua data juga berisi tentang informasi orientasi pen dan tekanan. *Task* pertama cocok untuk verifikasi tanda tangan dinamis pada perangkat-perangkat tablet yang tidak menyediakan informasi tekanan dan orientasi pen sedangkan *task* kedua cocok untuk perangkat yang menyediakan informasi lengkap dari koordinat, tekanan, dan orientasi pen. Setiap *task* terdiri dari 100 set tanda tangan. Setiap set tanda tangan terdiri dari 20 tanda tangan asli dari satu kontributor dan 20 tanda tangan palsu dengan kategori pemalsuan terlatih dari 5 kontributor yang lain. Tanda tangan diambil dengan menggunakan tablet WACOM. Hanya 40 set tanda tangan yang di-*release* kepada peserta dan tersedia secara online untuk mengembangkan dan mengevaluasi sistem verifikasi tanda tangan.

Setiap tanda tangan disimpan dalam teks yang berbeda (dengan ekstensi .txt). Konvensi penamaan dari file tanda tangan adalah UxSy, dimana x adalah ID user dan y adalah ID tanda tangan. File tanda tangan yang di-relese ke peserta memiliki nilai x dari 1 sampai 40. Tanda tangan asli mempunyai nilai y dari 1 sampai 20 dan tanda tangan dengan pemalsuan terlatih mempunyai nilai dari 21-40. Dalam file tanda tangan, tanda tangan direpresentasikan sebagai urutan poin. Baris pertama

adalah nilai integer yang menjelaskan total poin/titik dalam tanda tangan. Baris selanjutnya bersisi data-data per-poin dalam tanda tangan. Berikut adalah format yang disimpan dalam data task pertama:

- a Koordinat x = posisi kursor yang sudah di skalakan sepanjang x-axis
- b Koordinat y = posisi kursor yang sudah di skalakan sepanjang y-axis
- c Time stamp = waktu sistem saat sampling titik dilakukan
- d Button Status = status tombol (0 untuk pen-up dan 1 untuk pen-down)

2.6. Template Tanda Tangan

Template tanda tangan pengguna dibuat pada saat proses *enrollment* dimana sejumlah tanda tangan diambil dari pengguna dan kemudian fitur set dihitung dari setiap sampel. Setelah itu, variance setiap komponen fitur dihitung dan digunakan untuk membuat pengkuantisasi untuk setiap komponen fitur yang menghasilkan vektor kuantisasi step size Q_u . Vektor kuantisasi step size Q_u ini digunakan untuk mengkuantisasi setiap vektor fitur tanda tangan dari proses enrollment. Yang terakhir adalah nilai rata-rata dari vektor-vektor fitur yang telah dikuantisasi ini digunakan sebagai template \bar{F}^u . Satu pasang (Q_u, \bar{F}^u) yang terdiri dari vektor kuantisasi step size dan template vektor fitur kemudian disimpan yang kemudian digunakan untuk memverifikasi tanda tangan *online* pengguna u yang diklaim

Detail tentang bagaimana menurunkan vektor kuantisasi step size Q_u dan template vektor \bar{F}^u adalah sebagai berikut. Misalkan S adalah total sampel tanda tangan yang diambil pada saat proses enrollment dan M adalah jumlah total fitur untuk setiap sampel. Dan misalkan $FS_j = \{f_i^{S_j} | i = 1, \dots, M\}$ adalah vektor fitur dari sampel yang sudah di enroll s_j dari pengguna u dimana $j = 1, \dots, S$. vektor kuantisasi step size Q_u dari pengguna u , $Q_u = \{q_i^u | i = 1, \dots, M\}$, diperoleh dengan menghitung nilai standard deviasi dari semua sampel yang di enroll untuk setiap fitur.

$$q_i^u = \beta \sqrt{\frac{1}{S} \sum_{j=1}^S (f_i^{S_j} - \mu_{f_i^{(u)}})^2}, \quad i = 1, \dots, M \quad (2-12)$$

Dimana β adalah 1.5 yang ditentukan dengan eksperimen.

Kemudian, vektor fitur terkuantisasi dari setiap sampel yang dienroll s dari pengguna u , $\hat{F}^{(s|u)} = \{\hat{f}_i^{s_j} | i = 1, \dots, M\}$, dihitung dengan menggunakan vektor kuantisasi step size q_i^u dalam Q_u dengan rumus :

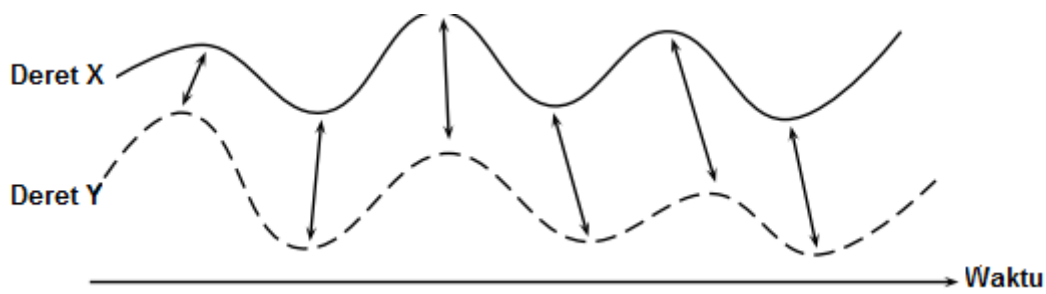
$$\hat{f}_i^{(s_j|u)} = \left\lfloor \frac{f_i^{s_j}}{q_i^u + \epsilon} \right\rfloor, \quad i = 1, \dots, M \quad (2-13)$$

Dimana ϵ adalah 0.0002 untuk histogram dengan frekuensi absolute dan 0.8 untuk frekuensi relatif yang tujuannya adalah mencegah pembagian oleh 0. Yang terakhir. Template vektor $\check{F}^u = \{\check{f}_i^u | i = 1, \dots, M\}$, diturunkan dengan menghitung rata-rata vektor fitur yang sudah dikuantisasi dari semua sampel tanda tangan online yang sudah di enroll dari pengguna u .

$$\check{f}_i^u = \left\lfloor \frac{\sum_{j=1}^S \hat{f}_i^{(s_j|u)}}{S} \right\rfloor, \quad i = 1, \dots, M \quad (2-14)$$

2.7. Dynamic Time Warping (DTW)

Dynamic Time Warping (DTW) adalah teknik yang sudah dikenal dengan baik untuk menemukan kesejajaran yang optimal antara dua deret (*time-dependent*) dalam batasan tertentu gambar 2.7. Secara intuisi, dua deret tersebut dilipat (*warped*) dengan cara *non-linear* untuk pencocokan satu sama lain. Pada awalnya, DTW digunakan untuk membandingkan pola *speech* yang berbeda dalam *automatic speech recognition* (Rabiner & Juang, 1993). Dalam bidang seperti *data mining* dan *information retrieval*, DTW diaplikasikan untuk mengatasi deformasi waktu dan perbedaan kecepatan pada data *time series*.



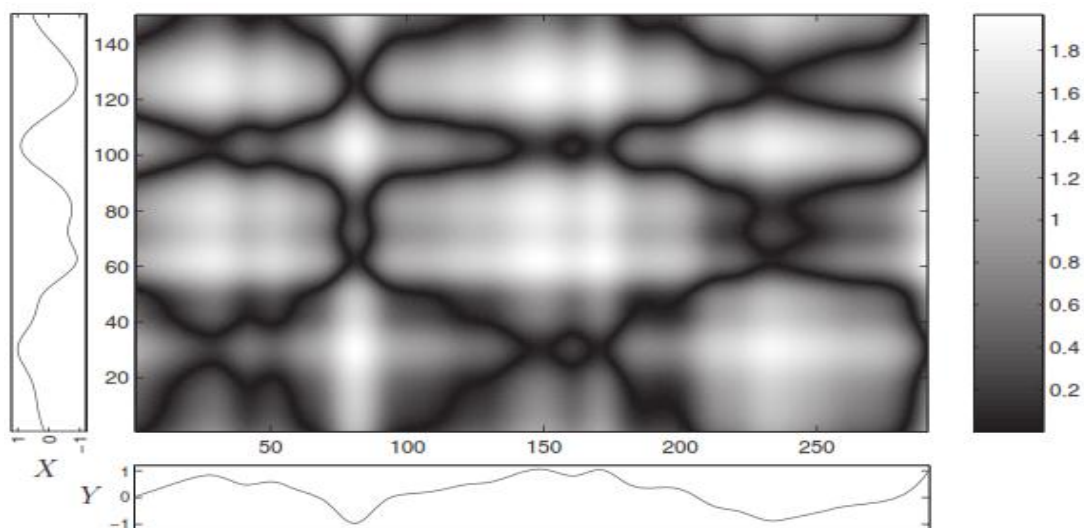
Gambar 2.12. Pensejajaran waktu dari dua deret. Titik yang disejajarkan ditunjukkan dengan panah

2.7.1. DTW Klasik

Tujuan dari DTW adalah untuk membandingkan dua deret (*time-dependent*) $X := (x_1, x_2, \dots, x_N)$ dengan panjang N dan $Y := (y_1, y_2, \dots, y_M)$ dengan panjang M . Deret-deret ini dapat berupa sinyal diskrit (*time series*). Berikut ini, kami menetapkan *feature space* yang dinotasikan dengan F . Dan $x_n, y_m \in F$ untuk $n \in [1: N]$ dan $m \in [1: M]$. Untuk membandingkan dua fitur berbeda $x, y \in F$, maka kita memerlukan pengukuran harga lokal (*local cost measure*) kadang-kadang juga disebut dengan pengukuran jarak lokal (*local distance measure*), yang didefinisikan sebagai berikut :

$$c: F \times F \rightarrow \mathbb{R}_{\geq 0} \quad (2-15)$$

Secara umum, $c(x, y)$ bernilai kecil (*low cost*) apabila x dan y mirip satu sama lain dan begitu juga sebaliknya. Menghitung *local cost measure* untuk setiap pasangan elemen deret X dan Y akan menghasilkan *cost matrix* $C \in \mathbb{R}^{N \times M}$ yang didefinisikan dengan $C(n, m) := c(x_n, y_m)$ lihat gambar 2.2.



Gambar 2.13. Cost Matrix dari dua deret X (vertikal aksis) dan Y (horizontal aksis)

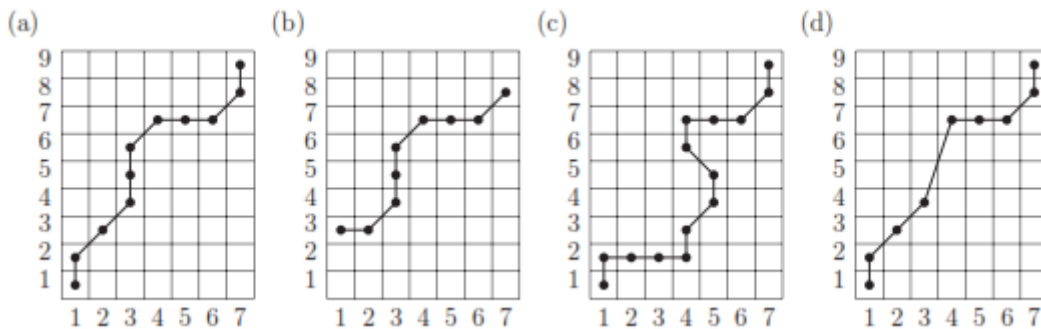
Gambar 2.2 merupakan gambar *cost matrix* dari dua deret bernilai *real* yaitu X (vertikal aksis) dan Y (horizontal aksis) yang menggunakan *manhattan distance* sebagai c (*local cost measure*). Daerah *low cost* diindikasikan dengan daerah dengan warna hitam sedangkan daerah *high cost* diindikasikan dengan warna putih.

Kemudian tujuan dari DTW adalah untuk menemukan kesajajaran antara X dan Y yang memiliki *cost* minimal secara keseluruhan. Secara intuisi, kesejajaran optimal seperti itu berada pada “lembah” dari *low cost* dalam *cost matrix*. Lihat figur 2.4 untuk ilustrasi. Selanjutnya kita akan memformulasikan notasi dari pensejajaran. Sebuah $(N-M)$ - *warping path* (atau secara lebih sederhana bisa disebut dengan *warping path* saja jika N dan M sudah jelas dalam konteks) adalah sebuah deret $p = (p_1, \dots, p_L)$ dengan $p_l = (n_l, m_l) \in [1:N] \times [1:M]$ untuk $l \in [1:L]$ memenuhi tiga kondisi berikut ini:

- a. Kondisi *Boundary* $p_1 = (1,1)$ dan $p_L = (N,M)$.
- b. Kondisi *monotocity* : $n_1 \leq n_2 \leq \dots \leq n_L$ dan $m_1 \leq m_2 \leq \dots \leq m_L$.
- c. Kondisi *step size*: $p_{l+1} - p_l \in \{(1,0), (0,1), (1,1)\}$ untuk $l \in [1:L - 1]$.

Yang perlu dicatat adalah kondisi *step size* (c) sudah mengimplikasikan kondisi *monotocity* namun tetap disebutkan secara eksplisit demi kejelasan. Sebuah (N,M) -*warping path* $p = (p_1, \dots, p_L)$ adalah titik-titik pensejajaran antara dua deret $X = (x_1, x_2, \dots, x_N)$ dan $Y = (y_1, y_2, \dots, y_M)$ dengan mengirimkan elemen x_{n_l} dari X kepada

elemen y_{m_l} dari Y . Kondisi *boundary* memastikan bahwa elemen pertama dan elemen terakhir dari X dan Y adalah sejajar satu sama lain. Kondisi *monotocity* mencerminkan kebutuhan terhadap waktu yang benar: Jika X digeser satu detik maka itu juga harus berlaku untuk elemen pasangannya di Y , dan begitu juga sebaliknya. Yang terakhir, kondisi *step size* mengekspresikan satu jenis kondisi *continuity*: tidak ada elemen dalam X dan Y yang dapat diabaikan dan tidak ada pengulangan dalam proses pensejajaran (dalam maksud bahwa semua pasangan indeks yang diisikan dalam *warping path* p adalah satu pasang yang berbeda). Gambar 2.3 mengilustrasikan tiga kondisi ini.



Gambar 2.14. Ilustrasi path dari pasangan indeks dari deret X dengan panjang $N=9$ dan deret Y dengan panjang $M=7$

Gambar *a* adalah contoh *warping path* yang memenuhi kondisi *boundary*, *monotocity*, dan *step size*. Gambar *b* adalah contoh *warping path* yang melanggar kondisi *boundary*. Gambar *c* adalah contoh *warping path* yang melanggar kondisi *monotocity*. Gambar *d* adalah contoh *warping path* yang melanggar kondisi *step size*. Total cost $c_p(X, Y)$ dari *warping path* p antara X dan Y dapat dihitung dengan rumus :

$$c_p(X, Y) := \sum_{l=1}^L c(x_{n_l}, y_{m_l}) \quad (2-16)$$

Optimal warping path antara X dan Y disimbolkan dengan p^* adalah *warping path* yang memiliki total *cost* minimal diantara semua kemungkinan *warping path*. Jarak DTW antara X dan Y adalah total cost dari p^* :

$$DTW(X, Y) := c_{p^*}(X, Y) \quad (2-17)$$

Selanjutnya beberapa sifat tentang jarak DTW. Pertama, jarak DTW tetap terdefinisi dengan baik, walaupun ada beberapa *warping path* dengan minimal *total cost*. Kedua, DTW bersifat simetrik dalam kasus *local cost measure* bersifat simetrik. Akan tetapi, jarak DTW secara umum tidak pasti positif bahkan juga untuk c . Sebagai contoh, kita dapat memperoleh $DTW(X, Y) = 0$ untuk deret $X := (x_1, x_2)$ dan $Y := (x_1, x_1, x_2, x_2, x_2)$ karena $c(x_1, x_1) = c(x_1, x_1) = 0$.

Untuk menentukan *optimal path* p^* , dapat dilakukan dengan mengetes setiap kemungkinan *warping path* antara X dan Y . Namun prosedur seperti itu, akan menyebabkan kompleksitas komputasi yang bersifat eksponensial terhadap panjang N dan M . Ada pendekatan yang lebih hemat komputasi dengan ukuran $O(NM)$ yaitu pendekatan *dynamic programming*. Untuk menjelaskan *dynamic programming*, kami mendefinisikan deretan prefix $X(1:n) := (x_1, \dots, x_n)$ untuk $n \in [1:N]$ dan $Y(1:m) := (x_1, \dots, x_m)$ untuk $m \in [1:M]$ dan juga persamaan :

$$D(n, m) := DTW(X(1:n), Y(1:m)) \quad (2-18)$$

$D(n, m)$ adalah nilai-nilai yang mengisi sebuah matrix D berdimensi $N \times M$, yang juga disebut dengan *accumulated cost matrix*. Tentunya, dalam matrix tersebut akan terdapat nilai $D(N, M) = DTW(X, Y)$. Sebuah tupel (n, m) , merepresentasikan sebuah entri dari *cost matrix* C atau dalam *accumulated cost matrix* D disebut dengan *cell*.

Selanjutnya kami akan menunjukkan bahwa D dapat dihitung secara efisien. *Accumulated cost matrix* D memenuhi identitas berikut $D(n, 1) = \sum_{k=1}^n c(x_k, y_1)$ untuk $n \in [1:N]$, $D(1, m) = \sum_{k=1}^m c(x_1, y_k)$ untuk $m \in [1:M]$, dan

$$D(n, m) = \min\{D(n-1, m-1), D(n-1, m), D(n, m-1)\} + c(x_n, y_m) \quad (2-19)$$

Teorema diatas memungkinkan komputasi rekursif dari matriks D . Pertama proses inisialisasi dilakukan dengan menambah matriks D dengan baris dan kolom

tambahan dan memberi nilai $D(n,0) := \infty$ dan $D(0,m) := \infty$. Selanjutnya proses rekursif dari (2-1) dapat dilakukan untuk $n \in [1 : N]$ dan $m \in [1 : M]$.

Lebih jauh lagi, matrix D dapat dihitung dengan pendekatan kolom dimana kolom ke- m hanya membutuhkan nilai dari kolom ke $(m-1)$. Hal tersebut mengimplikasikan apabila kita mencari nilai $DTW(X,Y)$, maka kebutuhan penyimpanan adalah $O(N)$. Serupa dengan hal itu, kita juga dapat *menghitung* dengan pendekatan baris dan akan menghasilkan kebutuhan penyimpanan sebesar $O(M)$. Akan tetapi, perlu dicatat bahwa *running time* adalah $O(NM)$ untuk kedua kasus tersebut. Untuk menghitung *warping path* yang optimal p^* , maka dibutuhkan keseluruhan matrix $D (N \times M)$.

Algoritma	: Optimal Warping Path
Input	: Accumulated cost matrix D
Output	: Optimal warping path p^*
Prosedur	: Optimal path $p^* = (p_1, \dots, p_L)$ dihitung dalam urutan yang terbalik, Indeks dimulai dengan $p_L = (N,M)$. Apabila nilai $(n,m) = (1,1)$, maka kita sudah mencapai $l = 1$ dan berhenti. Selain daripada itu :
$p_{l-1} = \begin{cases} (1, m-1), & \text{jika } n = 1 \\ (n-1, 1), & \text{jika } m = 1 \\ \operatorname{argmin} \left\{ \begin{matrix} D(n-1, m-1), D(n-1, m-1) \\ D(n, m-1) \end{matrix} \right\}, & \text{yang lain} \end{cases}$	
Apabila pasangan terkecil lebih dari satu, maka akan diambil secara leksikografi yang paling kiri	

2.7.2. Fast DTW

Dalam (Salvador & Chan, 2007), mereka mengenalkan algoritma *FastDTW* dimana mereka menemukan aproksimasi yang akurat dari *optimal warping path* antara dua *time series*. Algoritma *FastDTW* menghindari pendekatan *brute force* pada *dynamic programming* dari algoritma DTW yang orisinal dengan menggunakan pendekatan multilevel. Langkah awal, data *time series* di *down sampling* ke resolusi yang sangat rendah. *Warp path* ditemukan pada resolusi terendah dan kemudian “diproyeksikan” ke *time series* yang beresolusi lebih tinggi

secara inkremental. *Warp path* yang diproyeksikan, diperbaiki dan kemudian diproyeksikan lagi ke resolusi yang lebih tinggi. Proses perbaikan dan pemroyeksian diteruskan sampai ditemukan *warp path* untuk *time series* beresokusi utuh.

Pendekatan multilevel yang digunakan dalam algoritma *FastDTW* terinspirasi dari pendekatan multilevel yang digunakan dalam *graph bisection* (Karypis, Aggarwal, Kumar, & Shekhar, 1999). *Graph bisection* adalah kegiatan membagi sebuah graph kedalam porsi-porsi yang kurang lebih sama, sehingga jumlah tepi yang dibagi sekecil mungkin. Algoritma yang efektif dan efisien tersedia untuk graph yang kecil, akan tetapi, untuk graph yang besar, solusi yang ditemukan biasanya jauh dari optimal. Pendekatan multilevel dapat digunakan untuk menemukan solusi optimal untuk graph yang kecil, dan secara berulang mengembangkan graph dan “memperbaiki” solusi sebelumnya yang sudah didapatkan untuk permasalahan graph yang lebih besar.

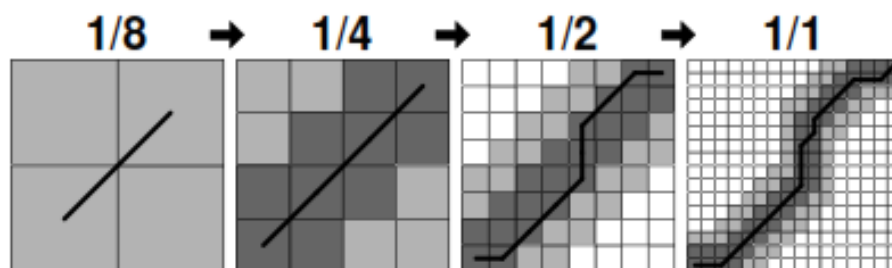
Algoritma *FastDTW* menggunakan pendekatan multilevel dengan tiga operasi kunci :

- a *Coarsening* : Menyusutkan *time series* menjadi *time series* yang lebih kecil yang merepresentasikan kurva yang sama dengan seakurat mungkin dengan titik-titik data yang lebih sedikit.
- b *Projection* : Menemukan jarak minimum dari *warp path* pada resolusi yang lebih rendah, dan menggunakan *warp path* tersebut sebagai tebakan awal untuk jarak *warp-path* minimum pada resolusi yang lebih tinggi.
- c *Refinement* : Memperbaiki *warp path* yang diproyeksikan dari resolusi yang lebih rendah dengan melakukan pengaturan lokal dari *warp path*.

Coarsening mengurangi ukuran / resolusi dari *time series* dengan merata-ratakan pasangan titik yang bersebelahan. *Time series* yang dihasilkan lebih kecil dengan faktor 2 dari *time series* yang asli. *Coarsening* dijalankan beberapa kali untuk menghasilkan banyak resolusi yang berbeda dari *time series*. *Projection* mengambil *warp path* yang dihitung pada resolusi yang lebih rendah dan menentukan pada *cell* mana yang dilewati oleh *warp path* dalam resolusi yang lebih tinggi selanjutnya. Oleh karena resolusi bertambah dengan faktor sama dengan 2, sebuah titik dalam *warp path* resolusi rendah akan menjadi paling tidak 4 titik pada resolusi yang lebih

tinggi. (kemungkinan $>4 |X| \neq |Y|$). Kemudian, *path* yang diproyeksikan ini digunakan sebagai heuristik selama perbaikan solusi untuk menemukan *warp path* pada resolusi yang lebih tinggi. *Refinement* menemukan *warp path* optimal di tetangga dari *path* yang diproyeksikan, dimana ukuran dari tetangga dikontrol oleh parameter *radius*. DTW klasik memiliki kompleksitas $O(N^2)$ karena setiap *cell* dalam *cost matrix* harus diisi untuk memastikan jawaban optimal ditemukan, dan ukuran matrix tumbuh secara kuadrat dengan ukuran *time series*. Dalam pendekatan multilevel, *cost matrix* hanya diisi pada tetangga *path* yang diproyeksikan dari resolusi sebelumnya. Oleh karena panjang dari *warp path* tumbuh secara linear dengan ukuran dari *time series* input, maka kompleksitas pendekatan multilevel adalah $O(N)$.

Pertama-tama algoritma *FastDTW* menggunakan *coarsening* untuk membuat semua resolusi yang akan dievaluasi. Figur 6 menunjukkan 4 resolusi yang dibuat saat menjalankan algoritma *FastDTW* pada *time series*. Algoritma DTW standard dijalankan untuk menemukan optimal *warp path* untuk resolusi terendah dari *time series*. *Warp path* pada resolusi terendah ditunjukkan pada gambar sebelah kiri dari gambar 6. Setelah *warp path* ditemukan untuk resolusi terendah, kemudian dia diproyeksikan ke resolusi yang lebih tinggi selanjutnya. Dalam figur 6, proyeksi *warp path* dari resolusi 1/8 ditunjukkan dengan cell berwarna lebih hitam pada resolusi 1/4.



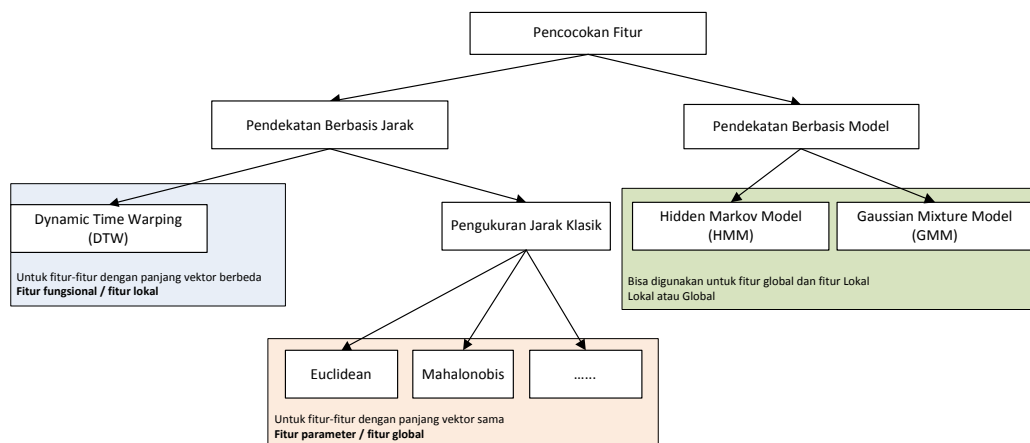
Gambar 2.15. Empat resolusi berbeda yang dievaluasi selama berjalanya algoritma *FastDTW*

Untuk memperbaiki *path* yang diproyeksikan, algoritma DTW yang sudah dibatasi dijalankan dengan batasan spesifik bahwa hanya *cells* dalam *warp path* yang

diproyeksikan yang dievaluasi. Ini akan menemukan *warp path* optimal melalui area dari *warp path* yang diproyeksikan dari resolusi yang lebih rendah. Akan tetapi, mungkin tidak keseluruhan *warp path* optimal masuk kedalam *path* yang diproyeksikan. Untuk meningkatkan kesempatan untuk menemukan solusi optimal, ada parameter *radius* yang mengendalikan jumlah cell tambahan pada setiap sisi dari *path* yang diproyeksikan. Dalam gambar 6, parameter *radius* diset menjadi 1. Cell yang termasuk selama perbaikan *warp path* oleh karena *radius* diwarnai hitam yang lebih terang. Ketika *warp path* diperbaiki pada resolusi 1/4, *warph path* tersebut diproyeksikan ke reolusi 1/2, diekspan dengan radius 1, dan kemudian diperbaiki lagi. Akhirnya, setelah *warph path* diproyeksikan ke matriks resolusi utuh (1/1) dalam gambar 6. Proyeksi diekspan dengan *radius* dan diperbaiki untuk terakhir kali. *Warph path* hasil perbaikan ini adalah keluaran dari algoritma.

2.8. Kategori Pencocokan Fitur

Dalam sesi ini, akan dijelaskan hubungan antara pendekatan verifikasi/pencocokan yang digunakan dengan jenis fitur yang diekstrak yang merepresentasikan tanda tangan. Berdasarkan (S Garcia-Salicetti et al., 2009) pendekatan dalam pencocokan fitur dibagi menjadi dua kelas yaitu pendekatan berbasis jarak (*distance-based approach*) dan pendekatan berbasis model (*model-based approach*).



Gambar 2.16. Kategori Pencocokan Fitur

2.8.1. Pendekatan Berbasis Jarak

Alternatif dalam pencocokan fitur berbasis jarak adalah dengan melakukan ekstraksi fitur global untuk membandingkan dua tanda tangan yang digambarkan oleh dua vektor dengan panjang yang sama. Jarak dari vektor tersebut dapat dihitung dengan penghitungan jarak klasik seperti Euclidean, Mahalanobis, dll. Dalam (S Garcia-Salicetti, 2007), ditunjukkan bahwa performansi sistem berdasarkan pengukuran klasik seperti di atas lebih lemah jika dibandingkan “elastic-distance” yang mencocokkan string karakter yang dihasilkan dari ekstraksi fitur yang kasar. Database yang digunakan dalam (S Garcia-Salicetti, 2007) adalah database MCYT.

Pada mulanya, Dynamic Time Warping digunakan secara eksklusif pada fungsi waktu yang ditangkap oleh digitizer atau tablet (tidak ada ekstraksi fitur yang dilakukan) dan secara terpisah dilakukan disetiap fungsi waktu. Contoh penggunaan dengan strategi seperti itu adalah pekerjaan yang dilakukan oleh (Komiya & Matsumoto, 1999) dan (Hangai, Yamanaka, & Hammamoto, 2000). Pada referensi (Komiya & Matsumoto, 1999) elastic distance antara tanda tangan uji dan tanda tangan referensi dihitung pada 3 jenis sinyal : koordinat, tekanan, dan sudut inklinasi pen (azimuth dan altitude) dan 3 skor yang dihasilkan digabungkan dengan menggunakan rata-rata yang berbobot. Database privat dengan ukuran kecil (8 orang) dan menggunakan 10 referensi tanda tangan, dihasilkan ERR dengan nilai 1.9%

Pada (Hangai et al., 2000), 3 elastic distance dihitung pada fungsi waktu yang dihasilkan oleh digitizer. Satu adalah koordinat bulat berasosiasi dengan 2 sudut inklinasi, satu lagi pada tekanan dan satu lagi adalah koordinat. Menjadi catatan bahwa pada studi ini, sudut inklinasi menjadi fungsi waktu yang paling baik ketika hanya satu elastic distance yang menjadi pertimbangan pada sebuah database pribadi milik autor. Hasil yang paling baik diperoleh saat penjumlahan digunakan untuk menggabungkan 3 elastic distance.

Sistem lain berdasarkan DTW yang melakukan pencocokan pada level berbeda selain level poin seperti yang disebutkan sebelumnya, juga diusulkan.

Salah satunya pada (Wirotius, Ramel, & Vincent, 2004) , penggabungan dari 3 *elastic distance* yang masing-masing dihasilkan dari mencocokkan 2 tanda tangan pada level deskripsi yang diberikan. Di sisi lain, sistem yang melakukan pencocokan pada level garis juga dilakukan (Chang & Shin, 2007) untuk mengurangi beban komputasi dari proses pencocokan.

(Wirotius et al., 2004) menggabungkan 3 *elastic distance* dengan rata-rata berbobot. Fitur fungsi waktu yang dibandingkan dari tanda tangan uji dan tanda tangan referensi adalah : temporal koordinat, panjang garis/trayektori/lintasan, koordinat. Sebelum pencocokan, data disiapkan terlebih dahulu dengan menyeleksi poin yang mewakili dalam tanda tangan, berkaitan dengan kecepatan minimum lokal.

Disisi yang lain (Chang & Shin, 2007) mengusulkan verifikasi tanda tangan berdasarkan DTW dan melakukan tes pada tandangan orang Jepang. Menarik untuk dicatat bahwa dalam tanda tangan Asia, informasi garis lebih representative daripada dalam tanda tangan Orang Barat. Metode yang diusulkan menggunakan DTW yang sudah dimodifikasi yang memungkinkan untuk penggabungan garis. Untuk mengendalikan proses ini dua aturan diusulkan: penalty-distance yang sesuai untuk mengurangi penggabungan garis dan batasan baru untuk menghindari penggabungan yang salah. Fungsi temporal (koordinat x dan y, tekanan, arah, dan altitude) serta informasi inter-stroke yang merupakan vektor dari titik pusat garis ke garis konsekuenya digunakan untuk pencocokan. Waktu total untuk tanda tangan juga digunakan sebagai fitur global untuk verifikasi.

Pekerjaan utama dalam kategori ini adalah milik (Kholmatov & Yanikoglu, 2005) dan (Jain et al., 2002), keduanya menggunakan Dynamic Time Warping. (Jain et al., 2002) melakukan pencocokan dengan mengkombinasikan jenis-jenis fitur yang berbeda. Mereka membuat sistem yang berbeda-beda berdasarkan fitur apa yang digunakan dan protokol mana yang digunakan (global threshold dan person-dependent threshold). Poin utama adalah fungsi biaya minimum bergantung pada perbedaan lokal dan juga karakteristik yang lebih global yang mendasarkan pada perbedaan jumlah garis. Resampling juga dilakukan tetapi hanya poin kritikal yang tetap (titik awal dan titik akhir garis dan titik perubahan trayektori). Fitur lokal

turunan dari posisi, sudut tangen, kecepatan relatif (kecepatan yang dinormalisasi oleh kecepatan rata-rata) adalah fitur terbaik dengan threshold global.

Sistem milik kolmatov (Kholmatov & Yanikoglu, 2005) adalah pendekatan yang memenangkan kompetisi pertama Online Signature Verification Competition (SVC) pada tahun 2004 (Yeung et al., 2004). Menggunakan turunan dari posisi sebagai 2 fitur lokal, dia mengkombinasikan DTW dengan skor normalisasi berdasarkan keberagaman intra-kelas dari klien, dihitung pada 8 tanda tangan yang digunakan untuk data latih. Pada 8 tanda tangan data latih ini, 3 faktor normalisasi dihasilkan dengan menghitung jarak DTW yang berpasangan dalam tanda tangan uji: nilai maksimum, nilai minimum, dan rata rata jarak.

2.8.2. Pendekatan Berbasis Model

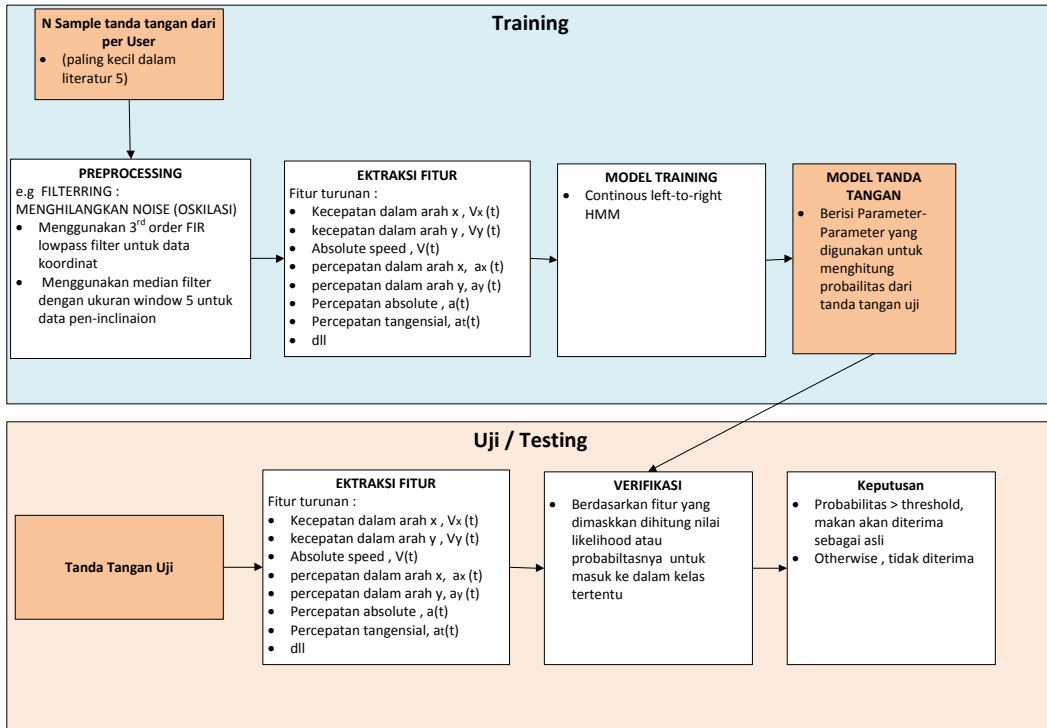
Pendekatan berbasis model muncul dengan wajar dalam verifikasi tanda tangan oleh karena Hidden Markov Models (HMMs) telah lama digunakan dalam pengenalan tulisan tangan. Satu dari pekerjaan pioner dan lengkap dalam literatur adalah pekerjaan milik (Dolfing, 1998; Dolfing, Aarts, & van Oosterhout, 1998). Dia memasang *continuous left-to-right* HMM dengan *Gaussian Mixture* dalam setiap state dengan fitur-fitur yang berbeda yang diekstrak pada level deskripsi tingkat intermediate, yaitu bagian dari tanda tangan yang didefinisikan oleh nol kecepatan vertikal. Dan juga, pentingnya setiap jenis fitur (spasial, dinamik, dan kontekstual) dipelajari dengan Linear Discriminant Analysis (LDA) pada database Philips. Fitur dinamik dan kontekstual muncul sebagai fitur yang lebih diskriminan dibandingkan dengan fitur spasial (Dolfing, 1998). Menggunakan 15 tanda tangan latih, dengan threshold yang bergantung pada person, ERR 1.9%-2.9% dicapai untuk setiap jenis pemalsuan.

Pada waktu yang sama, HMMs diskrit juga diproposalkan oleh (Kashi, Hu, Nelson, & Turin, 1998) dengan ekstraksi fitur lokal dengan hanya menggunakan sudut tangen dari path dan turunannya pada versi tanda tangan yang diresampling. Pengklasifikasi hybrid digunakan untuk mengambil keputusan dalam proses verifikasi: Pengklasifikasi yang lain menggunakan fitur global, berdasarkan jarak Mahalanobis dengan hipotesis dari fitur yang tidak berkorelasi dikombinasikan dengan kemungkinan HMM. Data latih lebih terbatas (6 data latih) dalam pekerjaan

ini. Performansi dievaluasi pada database Murray Hill berisi tanda tangan dari 59 subjek, menghasilkan EER 2.5 %. Kesimpulan dari pekerjaan ini adalah penggabungan skor dari dua pengklasifikasi menggunakan level deskripsi tanda tangan yang berbeda memberikan hasil yang lebih baik daripada hanya menggunakan HMM dengan fitur lokal.

HMM diskrit juga diproposalkan untuk tanda tangan online oleh (Rigoll & Kosmala, 1998) yang menggunakan jenis fitur yang berbeda antara lain: gambar resolusi rendah (“context bitmap”) disekitar titik lintasan, tekanan pen, sudut tangent lintasan dan turunannya, kecepatan, percepatan, beberapa fitur Fouriers, dan beberapa kombinasi dari fitur-fitur yang disebutkan sebelumnya. Performansi 99% dicapai dengan model ini dengan kombinasi dari fitur spasial dan fitur dinamis, namun sayangnya pada database milik pribadi yang tidak pernah digunakan oleh studi yang lain pada bidang ini.

Yang lebih baru, HMM kontinyu telah diproposalkan oleh (Fierrez, Ortega-Garcia, Ramos, & Gonzalez-Rodriguez, 2007) dengan menggunakan enkoding dinamis dari tanda tangan, mengeksploitasi fungsi waktu yang ditangkap oleh *digitizer* (koordinat x dan y, dan tekanan) ditambah sudut tangent lintasan, besar kecepatan, log curvature radius, total besar percepatan, dan turunan pertama mereka terhadap waktu sampai 14 fitur disetiap titik dari tanda tangan. Dalam tahap verifikasi, nilai kemungkinan (likelihood) diproses dengan menggunakan teknik skor-normalisasi yang berbeda digunakan dalam (Fierrez-Aguilar, Ortega-Garcia, & Gonzalez-Rodriguez, 2005). Hasil terbaik menggunakan subset dari database MCYT menghasilkan 0.78% EER untuk pemalsuan ber-skil (3.36% tanpa normalisasi skor)



Gambar 2.17. Alur pendekatan Berbasis Model

2.9. Pengukuran Performa

Pengukuran performa sistem verifikasi tanda tangan adalah pengukuran seberapa baik sistem tersebut dapat melakukan pencocokan dengan benar mengenai informasi tanda tangan dari orang yang sama dan menghindari pencocokan yang salah dari tanda tangan orang yang berbeda. Pada sistem verifikasi *confusion matrix* selalu digunakan sebagai acuan untuk mengukur kinerja sistem. Berdasarkan (Fawcett, 2006), berikut adalah *confusion matrix* yang merupakan matriks antara hasil prediksi dan aktual.

Tabel 2.2. Confusion Matrix

	Actual Positive	Actual Negative
Predict Positive	True Positive (TP)	False Positive (FP)
Predict Negative	False Negative (FN)	True Negative (TN)

Keterangan:

Tp (*True Positive*) : Jumlah klasifikasi yang benar dari data positif

Fp (*False Negative*) : Jumlah klasifikasi yang salah dari data negatif

Fn (*False Posotive*) : Jumlah klasifikasi yang salah dari data positif

Tn (*True Negative*) : Jumlah klasifikasi yang benar dari data negatif

2.9.1 Accuracy

Berdasarkan matriks *confusion*, maka dapat diketahui jumlah data dari masing-masing kelas yang diprediksi secara benar yaitu (TP dan TN) dan data yang diklasifikasi secara salah yaitu (FP dan FN). Dengan mengetahui jumlah data yang diklasifikasikan secara benar maka dapat diketahui akurasi hasil prediksi, dan dengan mengetahui jumlah data yang diklasifikasikan secara salah maka dapat diketahui laju *error* dari prediksi yang dilakukan. Untuk menghitung akurasi digunakan formula sebagai berikut:

$$\mathbf{Akurasi} = \frac{Tp+Tn}{Tp+Fp+Fn+Tn} \quad (2-20)$$

2.9.2 Precision dan Recall

Dalam bidang pencarian informasi, *precision* (disebut juga *positive prediction value*) merupakan metrik untuk mengukur kinerja sistem dalam mendapatkan data yang relevan. Sementara *recall* (disebut juga sensitivitas) merupakan metrik untuk mengukur kinerja sistem dalam mendapatkan data relevan yang terbaca (dalam bidang pencarian informasi). Dalam bidang penggalian data, *precision* adalah jumlah data yang *true positive* (jumlah data positif yang dikenali secara benar sebagai positif) dibagi dengan jumlah data yang dikenali sebagai positif, sedangkan *recall* adalah jumlah data yang *true positive* dibagi dengan jumlah data yang sebenarnya positif (*true positive + true negative*).

Mengacu pada tabel 2.4, berikut persamaan yang digunakan untuk menghitung *precision*:

$$\mathbf{Precision} = \frac{Tp}{Tp+Fp} \quad (2-21)$$

Persamaan yang digunakan untuk menghitung *recall* adalah:

$$Recall = \frac{Tp}{Tp+Fn} \quad (2-22)$$

2.9.3 F-Measure

F-measure merupakan parameter tunggal ukuran keberhasilan *retrieval* yang menggabungkan *precision* dan *recall* (Rijsbergen, 1979 dalam Arifiyanti, 2015).

$$FPR = \frac{(\beta^2 + 1)PR}{\beta^2 P + R} \quad (2-23)$$

Keterangan:

β = parameter kepentingan relatif aspek *precision* dan *recall*

P = nilai *precision*

R = nilai *recall*

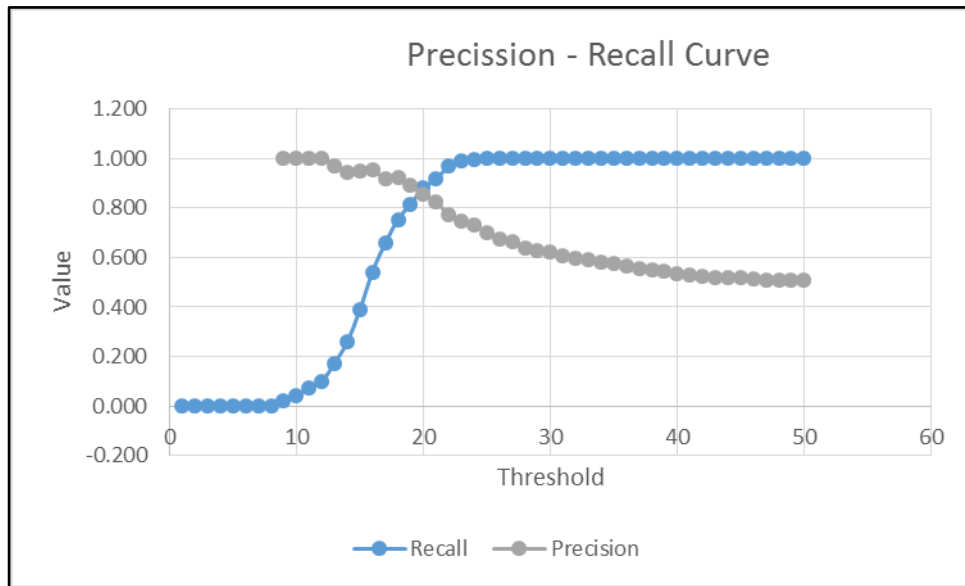
Jika nilai $\beta > 1$, maka akan memberikan bobot kepentingan *recall* lebih tinggi daripada *precision*. Jika nilai $\beta = 2$ maka bobot *recall* dua kali lebih besar daripada *precision*. Jika nilai $\beta = 0,5$ maka bobot *precision* dua kali lebih besar daripada *recall*. Tetapi jika *recall* dan *precision* memiliki bobot yang sama, maka $\beta = 1$ dan parameter *F-measure* dituliskan pada persamaan 2-24.

$$F - Measure = \frac{2PR}{P+R} \quad (2-24)$$

Suatu sistem klasifikasi dinyatakan efektif jika hasil perhitungan menunjukkan ketepatan (*precision*) yang tinggi sekalipun *recall*-nya rendah.

2.9.4. Precision-Recall Curve

Nilai Recall, precision, Accuracy dan F-Measure merupakan nilai pada satu titik threshold pada sistem verifikasi. Untuk menuntukan threshold yang baik digunakan Precision-Recall cuve dimana titik temu antara Recall dan Precission. Karena nilai Recall dan Precission adalah nilai yang saling bertolak belakang. Sumbu x merupakan threshold, dan sumbu y merupakan nilai Precision atau Recall nya. Berikut ilustrasi dari Precission dan Recall Curve :



Gambar 2.18. Kurva Preicision-Recall

2.9.5. Area Under Curve (AUC)

ROC *curve* atau kurva ROC digambarkan dalam grafik dua dimensi yang mengilustrasikan performa sistem verifikasi. Kurva tersebut dibuat dengan melakukan plotting antara nilai Recall atau *True Posotive Rate* (TPR) dan nilai *False Posotive Rate* (FPR) pada nilai threshold yang nilainya berbeda-beda. Rumus recall sudah diberikan pada sub-bab sebelumnya sedangkan berikut adalah rumus dari FPR:

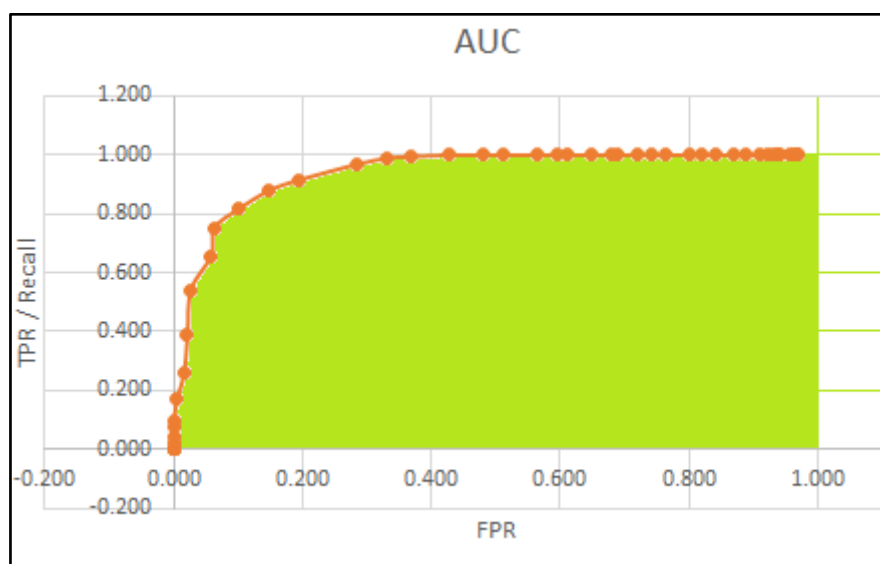
$$FPR = \frac{FP}{FP+TN} \quad (2-25)$$

Pada grafik ROC, sumbu y adalah nilai TPR dan sumbu x adalah nilai FPR. Kurva ROC menunjukkan hubungan antara TPR (disebut *sensitivity* atau *recall*) dengan FPR (disebut dengan *fall-out* atau *1-specificity*). *Area Under Curve* atau luas area dibawah kurva (AUC) merupakan representasi dari rata-rata sensitivity untuk semua nilai specificity yang mungkin. Nilai AUC ini digunakan untuk mengukur kinerja verifikasi secara umum karena merupakan nilai yang mempertimbangkan semua threshold. Nilai AUC memiliki rentang nilai antara 0 hingga 1, semakin mendekati 1 maka kinerja sistem verifikasi akan semakin baik.

Berikut adalah kategori kinerja berdasarkan nilai AUC nya (Fischer, Bachmann, & Jaeschke, 2003)

- 1.0 : verifikasi sempurna
- >0.9 : tingkat kinerja tinggi
- 0.71-0.9 : tingkat kinerja menengah
- 0.51-0.7 : tingkat kinerja rendah
- 0.5 : tingkat kinerja verifikasi acak

Berikut adalah ilustrasi dari AUC:



Gambar 2.19. Area Under Curve (AUC)

2.10. Pemeringkatan Fitur Berbasis Filter

Teknik pemeringkatan fitur berbasis filter melakukan pemeringkatan terhadap fitur secara independen tanpa menggunakan algoritma pembelajaran. Pemeringkatan fitur dilakukan dengan memberikan nilai kepada setiap fitur berdasarkan metode khusus tertentu. Dalam penelitian ini, beberapa teknik pemeringkatan fitur berbasis filter yang umum digunakan untuk mengetahui daftar peringkat fitur dalam tanda tangan dinamis antara lain *Chi Square*, *Information Gain*, *Gain Ratio*, *Symetrical Uncertainty*, dan *ReliefF*. Metode *Chi Square* (Plackett, 1983) digunakan untuk meneliti apakah tidak ada “asosiasi” antara dua fitur yang artinya apakah dua fitur tersebut independen. *Information Gain*, *Gain Ratio*, *Symetrical Uncertainty* adalah pengukuran berdasarkan konsep entropi

dimana berdasarkan teori informasi. *Information Gain* (Quinlan, 1986) adalah informasi yang menyediakan tentang atribut kelas target Y , diberikan nilai fitur independen X . *Information Gain* mengukur pengurangan rata-rata ketidakmurnian yang dibobot dari partisi, dibandingkan dengan ketidakmurnian dari data secara menyeluruh. Kelemahan dari *Information Gain* adalah dia cenderung terhadap fitur dengan jumlah kemungkinan nilai yang lebih besar. Satu strategi untuk mengatasi ini adalah dengan menggunakan *Gain Ratio* yang menghukum fitur multi-value. *Symmetrical Uncertainty* (SU) (Hall & Holmes, 2003) adalah cara lain untuk mengatasi masalah bias dari *Information Gain* terhadap fitur dengan nilai yang lebih banyak dengan membagi *Information Gain* dengan jumlah X dan Y . Relief adalah pemeringkatan fitur berbasis *instance* (Kira & Rendell, 1992), ReliefF adalah ekstensi dari algoritma Relief yang dapat mengatasi noise dan dataset multi kelas. Metode tersebut dilakukan dengan mengambil contoh *instance* secara acak dari data latih dan melakukan cek terhadap *instance* tetangga dari kelas yang sama dan juga dari kelas yang berbeda yang disebut dengan “near-hit” dan “near-miss”. Apabila “near-hit” memiliki nilai yang berbeda untuk atribut tertentu, maka atribut tersebut kemungkinan *irrelevant* dan untuk itu bobotnya dikurangi. Di sisi lain, apabila “near-miss” memiliki nilai yang sama untuk atribut tertentu, maka atribut tersebut kemungkinan relevan dan untuk itu bobotnya ditambahi. Ini merupakan prosedur standard yang digunakan untuk pembobotan fitur untuk pembelajaran berbasis *instance*.

2.10.1. Information Gain

Diberikan entropi adalah kriteria ketidakmurnian dalam data latih S , kita dapat mendefinisikan sebuah pengukuran yang merefleksikan informasi tambahan tentang Y yang disediakan oleh X yang merepresentasikan jumlah dimana entropi Y berkurang. Ini diberikan oleh rumus :

$$IG = H(Y) - H(Y|X) = H(X) - H(X|Y) \quad (2-26)$$

IG adalah pengukuran simetris. Informasi yang diperoleh tentang Y setelah mengobservasi X adalah sama dengan informasi yang diperoleh tentang X setelah mengobservasi Y . Kelemahan dari IG adalah kebiasaan dalam kecenderungan dengan fitur yang memiliki nilai lebih banyak bahkan apabila mereka tidak informatif

2.10.2. Gain Ratio

Gain ratio adalah pengukuran non-simetris yang diperkenalkan untuk mengkompensasi bias dari Information Gain [31]. GR diberikan oleh :

$$GR = \frac{IG}{H(X)} \quad (2-27)$$

Dengan persamaan diatas, ketika variabel Y harus diprediksi, maka IG dinormalisasi dengan membagi dengan entropi dari X , dan juga sebaliknya. Oleh karena normalisasi ini, nilai GR selalu berada dalam batas $[0,1]$. Sebuah nilai $GR = 1$ mengindikasikan bahwa pengetahuan dari X secara lengkap memprediksi Y , dan $GR = 0$ berarti bahwa tidak ada relasi antara Y dan X . Berlawanan dengan IG, GR lebih menyukai fitur dengan nilai yang lebih sedikit.

2.10.3. Relief

Misalnya masalah klasifikasi dua kelas. Sebuah *instance* direpresentasikan oleh vektor yang dibentuk oleh sejumlah fitur nilai p . S mendefinisikan satu set instance latihan dengan ukuran n . F adalah fitur-fitur set (f_1, f_2, \dots, f_p) . Sebuah *instance* X didefinisikan oleh vektor berdimensi p (x_1, x_2, \dots, x_p) dimana x_j mendenotasikan nilai dari fitur f_j dari X . *Relief* adalah algoritma yang terinspirasi oleh pembelajaran berbasis *instance* (Aha, Kibler, & Albert, 1991), (Callan, Fawcett, & Rissland, 1991). Diberikan data latihan S , ukuran sampling m dan *threshold* relevansi τ . *Relief* mendeteksi fitur-fitur yang secara statistik relevan terhadap target. τ mendeskripsikan batasan relevansi ($0 \leq \tau \leq 1$). Skala setiap fitur diasumsikan oleh $[\]$ adalah nominal (termasuk boolean) atau numerik (integer atau real). Perbedaan nilai fitur antara dua *instance* X dan Y didefinisikan oleh fungsi *diff* sebagai berikut. Saat x_k dan y_k adalah nominal,

$$diff(x_k, y_k) = \begin{cases} 0, & \text{jika } x_k \text{ dan } y_k \text{ sama} \\ 1, & \text{jika } x_k \text{ dan } y_k \text{ berbeda} \end{cases} \quad (2-28)$$

Saat x_k dan y_k adalah numerik

$$diff(x_k, y_k) = (x_k - y_k) / nu_k \quad (2-19)$$

dimana nu_k , adalah unit untuk menormalisasi nilai $diff$ kedalam interval $[0,1]$. *Relief* mengambil sebuah sample yang dibuat oleh *instance X*, *instance near-hit* dari *instance X*, dan *instance near-miss* dari *instance X*. *Relief* menggunakan *euclidean distance* berdimensi p untuk memilih *near-hit* dan *near-miss*. *Relief* memanggil fungsi untuk melakukan update bobot vektor W untuk setiap triplet dan menentukan rata-rata bobot fitur vektor *Relevance* (dari semua fitur kepada target). Tingkat *relevance* besar untuk fitur relevan dan kecil untuk fitur yang tidak relevan.

2.10.4. Correlation

Korelasi (Pearson Correlation) adalah parameter atau nilai untuk mengukur hubungan antara atribut dengan kelasnya. Dengan nilai-nilai ini kita dapat memeriksa seberapa kuat fitur-fitur tertentu terkait secara linier ke kelas. Nilai antara -1 dan 1. Nilai 0 menunjukkan tidak ada hubungan antara fitur dan kelas. Fitur yang bagus adalah fitur yang memiliki nilai absolut yang besar. Korelasi Pearson dapat dihitung dengan persamaan berikut:

$$\rho(X, Y) = \frac{\text{cov}(X, Y)}{\sigma_x \sigma_y} \quad (2-30)$$

Keterangan:

$\text{cov}(X, Y)$:	covariance between X and Y
σ_x	:	standard deviation of X
σ_y	:	standard deviation of Y

(lembar ini sengaja dikosongkan)

BAB 3

METODOLOGI PENELITIAN

Dalam bab ini diuraikan tahap-tahap yang akan dilakukan pada penelitian ini. Secara umum, tahapan penelitian ini dapat dilihat pada gambar 3.1. Tahapan identifikasi masalah, perumusan masalah, dan penentuan tujuan, ruang lingkup, dan kontribusi penelitian sudah dijelaskan dalam bab pendahuluan. Sedangkan tahap studi literatur yang mencakup landasan teori dan kajian pustaka telah dijelaskan pada bab 2. Pada bab 3 ini, akan dijelaskan mengenai tahap Penyiapan Data, Praproses Data, Derivasi Data, Ekstraksi Fitur Histogram, Pembuatan Template Tanda Tangan Pengguna, Skenario Percobaan dan Analisis Hasil, Penyusunan Kesimpulan, dan Saran Pengembangan Penelitian.

Gambar 3.1 menjelaskan metodologi penelitian secara umum yang terdiri dari beberapa tahap, yaitu identifikasi masalah, studi literatur, perumusan masalah, penetapan tujuan, ruang lingkup dan kontribusi penelitian, penyiapan data, praproses data, derivasi data, ekstraksi fitur histogram, pembuatan *template* tanda tangan pengguna, pencocokan dengan Dynamic Time Warping, analisis hasil, serta penyusunan kesimpulan dan saran pengembangan penelitian.

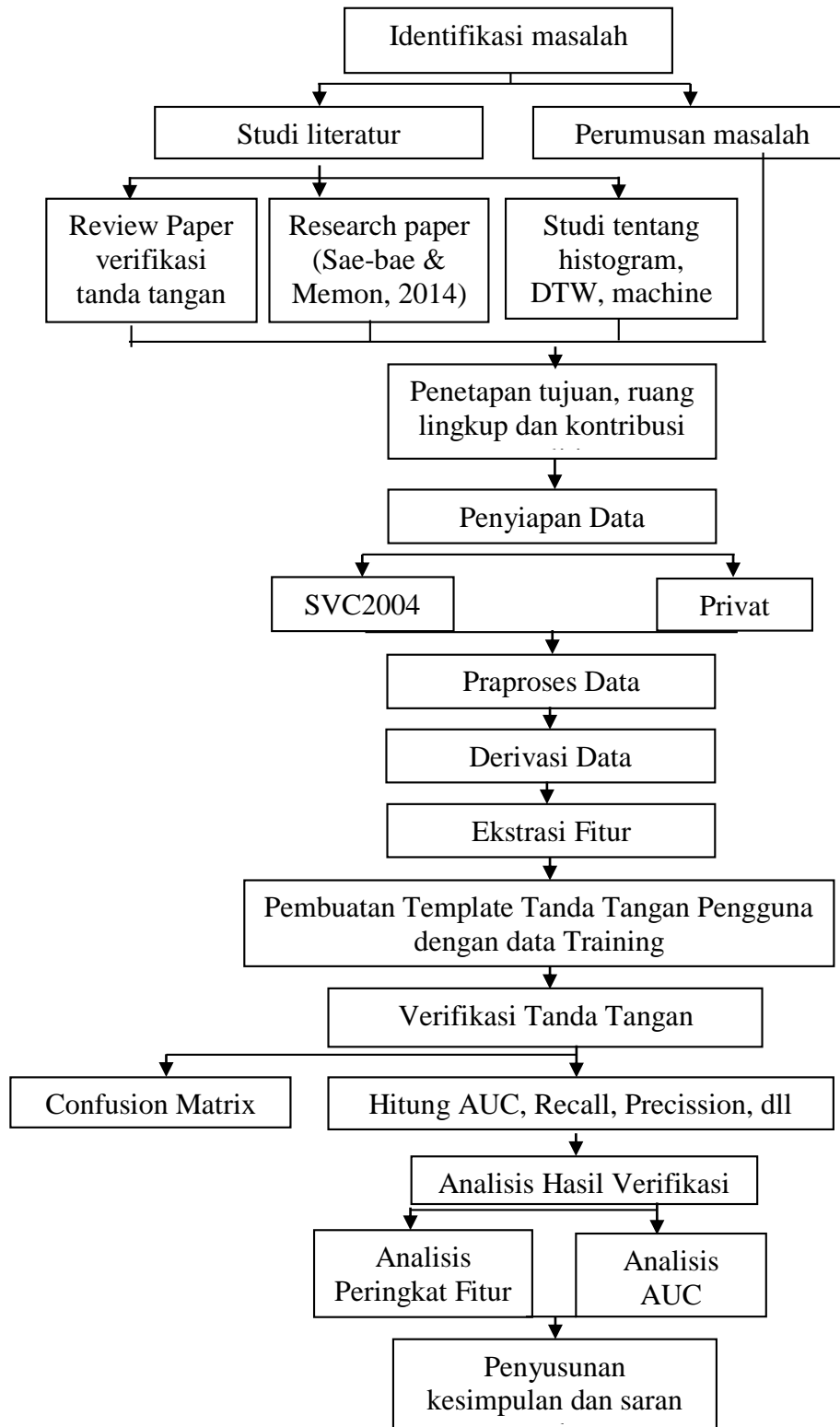
3.1. Penyiapan Data

Dalam penelitian ini, ada beberapa dataset tanda tangan yang digunakan dimana data tersebut tersedia di Internet yang bisa digunakan untuk penelitian sistem verifikasi tanda tangan *online* yaitu dataset SVC2004. Selain menggunakan dataset publik, dalam penelitian ini juga menggunakan data yang dibuat sendiri dengan menyesuaikan format data yang ada. Berikut penjelasan masing-masing dataset

3.1.1. Dataset Privat

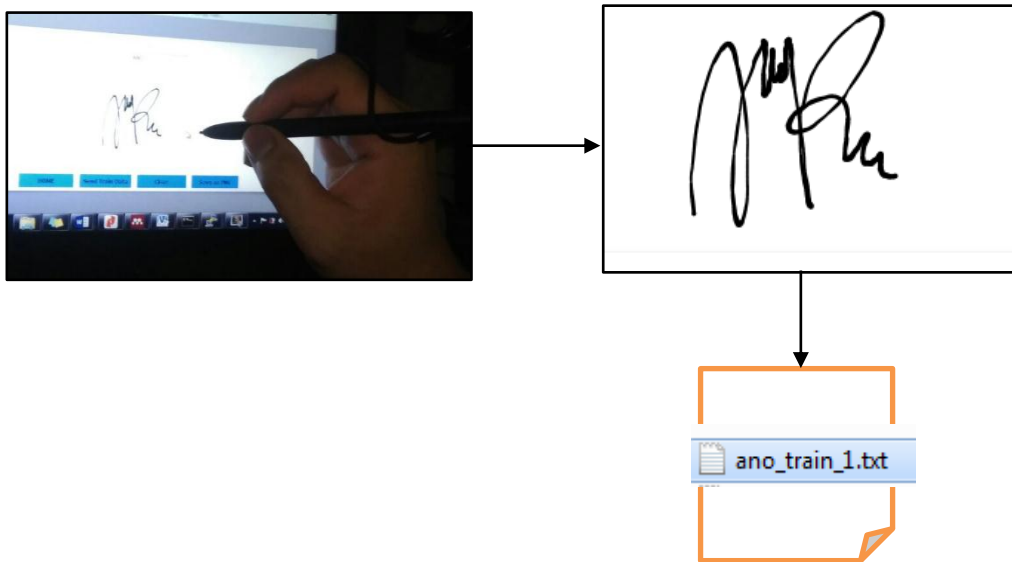
Tanda tangan diambil menggunakan aplikasi Web dengan teknologi HTML5. Aplikasi dibangun didalam notebook dan diakses melalui LAN yang dihubungkan dengan menggunakan modem wifi. Format data yang diambil adalah posisi x, y, dan *time stamp* sesuai dengan format data yang diambil pada perangkat

mobile seperti pada penelitian (Sae-bae & Memon, 2014), dimana informasi tekanan dan sudut orientasi tidak digunakan.



Gambar 3.1 Metodologi Penelitian

Tanda tangan diakuisisi dengan menggunakan perangkat *mobile Coolpad R18* dengan sistem operasi Android 5.1.1. Jumlah orang yang diambil tanda tanganya adalah 19 orang. Masing-masing orang diambil 20 tanda tangan asli dan 10 tanda tangan palsu yang dibuat oleh orang lain. Tanda tangan palsu dibuat dengan melihat video rekaman pada saat pembuatan tanda tangan asli. Semua tanda tangan tersebut disimpan dalam file text di server untuk pemrosesan lebih lanjut. Ilustrasi pengambilan data tanda tangan dapat dilihat dalam gambar berikut :



Gambar 3.2 Proses Enrollement Data Tanda Tangan

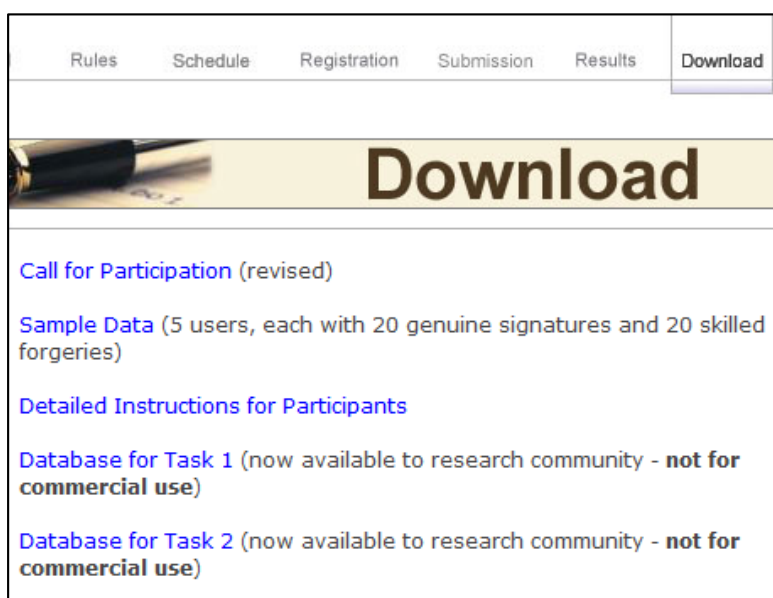
Tabel 3.1 Contoh 5 titik pertama hasil *enrollment*

Baris	x	y	timeStamp
1	0.000000000000e+00	0.000000000000e+00	1.504934990100e+12
2	0.000000000000e+00	1.000000000000e+00	1.504934990168e+12
3	1.000000000000e+00	7.000000000000e+00	1.504934990178e+12
4	6.000000000000e+00	1.600000000000e+01	1.504934990188e+12
5	2.100000000000e+01	3.400000000000e+01	1.504934990203e+12

Contoh titik di atas menunjukkan hasil *enrollment* tanda tangan yaitu posisi *x*, *y* dan *time stamp*. *Timestamp* yang digunakan adalah waktu dalam mili seconds sejak pertama kali waktu *unix* berlangsung atau disebut dengan *epoch time* (1 Januari 1970).

3.1.2. Dataset SVC2004

Pejelasan teori dataset SVC2004 dapat dilihat dalam bab 2. Dataset SVC2004 dapat diunduh di <http://www.cse.ust.hk/svc2004/download.html>. Semua orang dapat mengakses dataset tersebut. Dalam halaman tersebut akan tersedia dua *link* untuk mengunduh *database task 1* dan *database task 2*. Dalam penelitian ini semua informasi pada *database task 1* akan digunakan (posisi x, y, dan *timestamp*). Sedangkan informasi pada *database 2* yang digunakan adalah (posisi x, y, tekanan p, dan *timestamp*). File disimpan dalam bentuk *Task1.zip* dan *Task2.zip*



Gambar 3.3 Halaman unduh dataset SVC2004

3.2 Praproses Data

Dataset SVC2004 memiliki karakteristik y dimana *time stamp* memiliki interval yang sama karena data sudah dinormalisasi. Berbeda dengan dataset SVC2004, dataset privat memiliki perbedaan selisih waktu dari satu titik ke titik yang lain, yang tergantung pada ketersediaan sumber daya komputasi. Selain itu semua tanda tangan baik dataset publik dan privat memiliki variasi jumlah garis untuk setiap pengguna. Oleh karena itu, perlu dilakukan praproses data tanda tangan terlebih dahulu sebelum melalui proses selanjutnya.

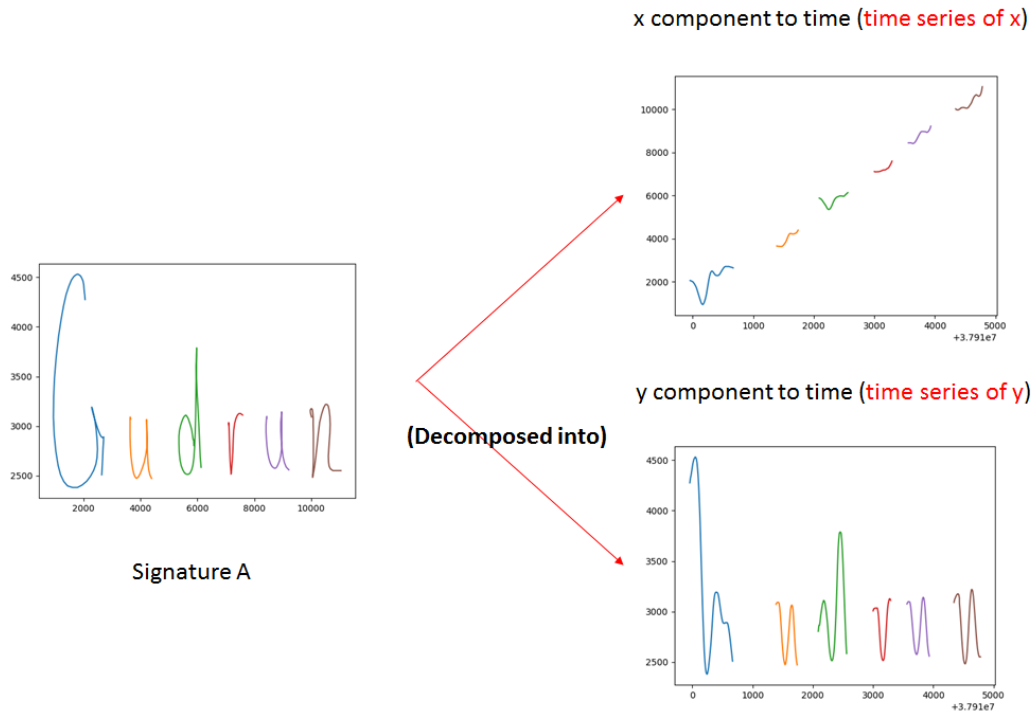
3.2.1. Normalisasi Waktu

Untuk dataset privat, praproses data yang perlu dilakukan adalah Normalisasi waktu. Normalisasi waktu digunakan dengan tujuan untuk menghasilkan tanda tangan online yang memiliki selisih interval dari satu titik ke titik yang lain dalam waktu yang sama. Simulasi proses yang dilakukan adalah sebagai berikut : Misalkan $S = \{v_1, v_2, \dots, v_N\}$ adalah tanda tangan *online* dengan jumlah garis N dimana setiap garis $v_i = \{(x_1^i, y_1^i), \dots, (x_M^i, y_M^i)\}$ adalah deret titik, yang disampling pada waktu $T = \{t_i^1, \dots, t_M^1\}$. Garis yang telah dinormalisasi s_i dihitung dengan menginterpolasi garis v_i pada $T = \left\{t_i^1, t_i^1 + R, t_i^1 + 2R, \dots, t_i^1 + \frac{\lfloor t_M^1 - t_i^1 \rfloor}{R} R\right\}$. Setelah proses normalisasi waktu, semua tanda tangan memiliki kecepatan sampling yang sama.

3.2.2. Penggabungan Garis

Penggabungan dilakukan untuk semua dataset, baik dataset publik (SVC2004) dan dataset privat. Untuk mengatasi variasi jumlah garis, garis-garis tanda tangan digabung sebelum verifikasi. Misalkan $S = \{s_1, s_2, \dots, s_N\}$ adalah tanda tangan *online* dengan deretan dari garis yang sudah dinormalisasi dimana setiap garis $s_i = \{(x_1^i, y_1^i), \dots, (x_M^i, y_M^i)\}$ adalah deret dari titik (x_j, y_j) dengan panjang M . Garis selanjutnya digabung dengan cara mentranslasi titik origin dari garis tersebut ke titik akhir dari garis sebelumnya.

3.3. Derivasi Data



Gambar 3.4 Dekomposisi tandan tangan dinamis menjadi *time series x* dan *time series y*

Gambar diatas adalah salah satu contoh tanda tangan dinamis yang dipecah menjadi dari dua buah *time series* yaitu *time series x* dan *time series y*. Setelah kedua *time series* tersebut selesai dinormalisasi, langkah selanjutnya adalah derivasi data atau penurunan data. Seperti yang dijelaskan sebelumnya, bahwa informasi yang tersedia adalah posisi x, y . Himpunan dari posisi x dan y dinotasikan dengan (X, Y) . Sebelum melakukan proses derivasi data, diperlukan penghitungan untuk mendapatkan informasi implisit dari tanda tangan yang antara lain adalah θ yaitu sudut yang terbentuk pada koordinat (x, y) dan juga r yaitu panjang garis antara titik (x, y) dengan titik origin $(0, 0)$. Dan himpunan dari θ dan r dinotasikan dengan Φ dan R . Sehingga informasi lengkap yang didapatkan sebelum proses derivasi data adalah (X, Y, Φ, R) .

Setelah informasi lengkap, masing-masing vektor akan diderivasi sampai tingkat ke-2. Derivasi data yang dilakukan disini adalah derivasi data dengan metode pendekatan yang sama dengan (Sae-bae & Memon, 2014) misalnya dari data x_n diperoleh x_n^I dengan persamaan $x_n^I = x_n - x_{(n-1)}$. Hasil dari derivasi tingkat

satu adalah (X^1, Y^1, Φ^1, R^1) sedangkan hasil derivasi tingkat dua adalah (X^2, Y^2, Φ^2, R^2) . Vektor – vektor hasil derivasi inilah yang akan digunakan untuk ekstraksi fitur histogram pada langkah selanjutnya. Untuk lebih jelasnya misalnya adalah contoh himpunan X sederhana pada tabel di bawah ini.

Tabel 3.2 Contoh derivasi deret posisi atau Himpunan X

Himpunan	t	t + Δt	t + 2 Δt	t + 3 Δt	t + 4 Δt	t + 5 Δt
X	5	7	3	9	8	10
X^1	2	-4	6	-1	2	-
X^2	-6	10	-7	3	-	-

Dapat dilihat bahwa dari satu tingkat diturunkan ke tingkat yang selanjutnya panjang himpunan akan berkurang satu, X memiliki panjang 6, X^1 memiliki panjang 5, dan X^2 memiliki panjang 4.

3.4. Ekstraksi Fitur Histogram

Sesuai dengan yang dijelaskan pada bab satu bahwa penelitian ini bertujuan untuk membandingkan sistem yang dibangun dengan menambahkan fitur-fitur baru yang belum digunakan dalam penelitian (Sae-bae & Memon, 2014) dengan sistem yang dibangun tanpa menambahkan fitur-fitur baru tersebut. Fitur-fitur yang digunakan tanpa ada penambahan fitur-fitur yang baru adalah fitur-fitur yang diambil dari histogram-histogram sebagai berikut :

Tabel 3.3 Daftar Histogram Asli

No	Histogram	Atribut Input	Min	Max	Bins
1	Φ^1	$\{\theta_1^1, \dots, \theta_n^1\}$	$-\pi$	π	24
2	Φ^2	$\{\theta_1^2, \dots, \theta_n^2\}$	$-\pi$	π	24
3	$\langle \Phi^1, \Phi_{d(1,2)}^1 \rangle$	$\{\theta_1^1, \dots, \theta_{n-1}^1, \theta_1^1, \dots, \theta_{n-2}^1\}$ $\{\theta_2^1, \dots, \theta_n^1, \theta_3^1, \dots, \theta_n^1\}$	$-\pi$	π	8 8
4	R^1	$\{r_1^1, \dots, r_n^1\}$	0	$\mu+3\sigma$	16
5	R^2	$\{r_1^2, \dots, r_n^2\}$	0	$\mu+3\sigma$	16
6	X^1	$\{x_1^1, \dots, x_n^1\}$	$\mu-3\sigma$	$\mu+3\sigma$	8
7	Y^1	$\{y_1^1, \dots, y_n^1\}$	$\mu-3\sigma$	$\mu+3\sigma$	8
8	X^2	$\{x_1^2, \dots, x_n^2\}$	$\mu-3\sigma$	$\mu+3\sigma$	8
9	Y^2	$\{y_1^2, \dots, y_n^2\}$	$\mu-3\sigma$	$\mu+3\sigma$	8
10	$\langle X^1, X^2 \rangle$	$\{x_1^1, \dots, x_n^1\}$ $\{x_1^2, \dots, x_n^2\}$	$\mu-3\sigma$ $\mu-3\sigma$	$\mu+3\sigma$ $\mu+3\sigma$	6 4
11	$\langle Y^1, Y^2 \rangle$	$\{y_1^1, \dots, y_n^1\}$ $\{y_1^2, \dots, y_n^2\}$	$\mu-3\sigma$ $\mu-3\sigma$	$\mu+3\sigma$ $\mu+3\sigma$	6 4

12	$\langle \Phi^1, R^1 \rangle_{(1)}$	$\{\theta_1^1, \dots, \theta_{[n/2]}^1\}$	$-\pi$	Π	8
		$\{r_1^1, \dots, r_{[n/2]}^1\}$	0	$\mu+3\sigma$	4
	$\langle \Phi^1, R^1 \rangle_{(2)}$	$\{\theta_{[n/2]}^1, \dots, \theta_n^1\}$	$-\pi$	Π	8
		$\{r_{[n/2]}^1, \dots, r_n^1\}$	0	$\mu+3\sigma$	4
13	$\langle \Phi^2, R^2 \rangle_{(1)}$	$\{\theta_1^2, \dots, \theta_{[n/2]}^2\}$	$-\pi$	Π	8
		$\{r_1^2, \dots, r_{[n/2]}^2\}$	0	$\mu+3\sigma$	4
	$\langle \Phi^2, R^2 \rangle_{(2)}$	$\{\theta_{[n/2]}^2, \dots, \theta_n^2\}$	$-\pi$	Π	8
		$\{r_{[n/2]}^2, \dots, r_n^2\}$	0	$\mu+3\sigma$	4
14	$\langle \Phi^1, R^2 \rangle_{(1)}$	$\{\theta_1^1, \dots, \theta_{[n/2]}^1\}$	$-\pi$	π	8
		$\{r_1^2, \dots, r_{[n/2]}^2\}$	0	$\mu+3\sigma$	4
	$\langle \Phi^1, R^2 \rangle_{(2)}$	$\{\theta_{[n/2]}^1, \dots, \theta_n^1\}$	$-\pi$	π	8
		$\{r_{[n/2]}^2, \dots, r_n^2\}$	0	$\mu+3\sigma$	4

Sedangkan fitur-fitur yang baru diturunkan dari dua histogram yang dijelaskan dalam tabel di bawah ini :

Tabel 3.4 Daftar Histogram yang ditambahkan

No	Histogram	Atribut Input	Min	Max	bins
1	X	$\{x_1, \dots, x_n\}$	0	$\mu+3\sigma$	8
2	Y	$\{y_1, \dots, y_n\}$	0	$\mu+3\sigma$	8

Dapat dilihat bahwa jumlah bin berbeda bergantung pada jenis variabelnya. Jumlah bin yang ditulis diatas sesuai dengan (Sae-bae & Memon, 2014). Begitu juga dengan jumlah bin pada histogram baru, diambil nilai 8 karena variabel yang sifatnya diskrit seperti itu pada referensi jumlah bin-nya juga 8. Sedangkan variabel yang sifatnya halus / pecahan seperti sudut dan panjang garis dibuat jumlah bin yang lebih banyak. Semakin halus jumlah bin dibuat lebih besar dengan tujuan supaya lebih bisa menangkap perbedaan nilai yang kecil dalam bentuk pecahan dan dibedakan dalam bin yang berbeda sehingga akan menghasilkan fitur yang lebih baik. Histogram-histogram diatas dihitung dengan membagi nilai atribut yang dibatasi oleh nilai *min* dan *max* kedalam beberapa jumlah bin dengan lebar yang sama dan kemudian menghitung semua elemen yang masuk ke dalam setiap bin tertentu. Untuk atribut sudut dan turunanya, jarak histogramnya adalah $[-\pi, \pi]$. Untuk atribut yang tidak memiliki batasan secara eksplisit, maka dilakukan proses *cutoff* sebanyak tiga kali standard deviasi dari rata-rata untuk mengasilkan batasan

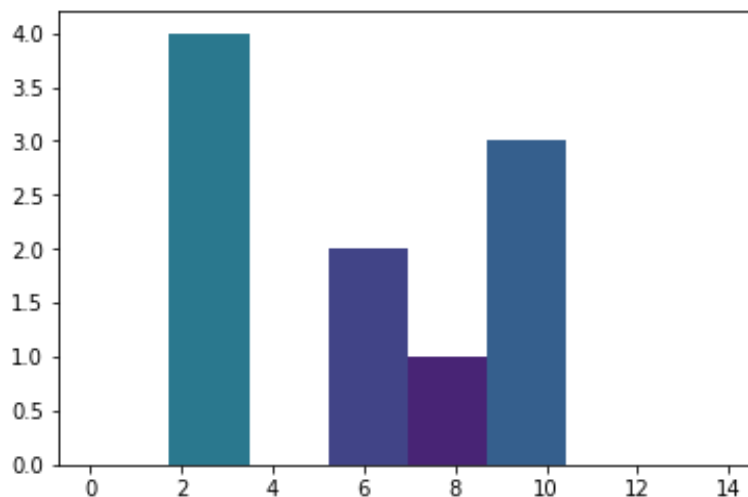
seperti yang digambarkan dalam tabel. Frekuensi relatif dihitung disetiap bin untuk digunakan sebagai fitur. Frekuensi relatif adalah jumlah item yang masuk disetiap bin dibagi dengan jumlah total item dalam himpunan atau darat tersebut. Setelah semua frekuensi relatif setiap histogram dihitung, mereka semua digabung dan digunakan sebagai vektor fitur. Misalnya B_i adalah sebuah vektor dari frekuensi bin dari histogram ke-i. Fitur vektor F didefinisikan sebagai $F = \{ B_1 \parallel B_2 \parallel \dots \parallel B_j \}$, dimana j adalah jumlah total histogram.

Sebagai ilustrasi, misalkan deret atau himpunan $X = \{9, 6, 3, 3, 6, 2, 2, 9, 7, 9\}$. Maka range histogram dari X adalah $(0 - 13.89216498)$. Nilai min untuk X otomatis adalah 0 dan nilai maksimum dihitung dari $(\mu+3\sigma)$. Histogram range dan frekuensi yang terbentuk dari 8 bins adalah seperti pada tabel berikut :

Tabel 3.5 Contoh Histogram X Dalam Bentuk Tabel

No	Bin 1	Bin 2	Bin 3	Bin 4	Bin 5	Bin 6	Bin 7	Bin 8
Range	0 - 1.736	1.736 - 3.473	3.473 - 5.209	5.209 - 6.946	6.946 - 8.682	8.682 - 10.419	10.419 - 12.155	12.155 - 13.892
Frekuensi	0	4	0	2	1	3	0	0
Frekuensi Relatif	0	0.4	0	0.2	0.1	0.3	0	0

Dan berikut adalah gambar histogram yang terbentuk dari informasi yang dicontohkan di dalam tabel diatas:



Gambar 3.5 Contoh Histogram X digambarkan dalam bentuk grafik

Selanjutnya fitur yang diambil adalah frekuensi relatif masing-masing bin, dalam kasus ini maka histogram X akan menghasilkan vektor fitur $\{0, 0.4, 0, 0.2, 0.1, 0.3, 0, 0\}$,

0, 0}. Semua tanda tangan dalam masing-masing dataset diekstrak fitur histogramnya yang disimpan dalam format .txt.

3.5. Pembuatan Template Tanda Tangan Pengguna

Template tanda tangan pengguna dibuat pada saat proses *enrollment* dimana sejumlah tanda tangan diambil dari pengguna dan kemudian fitur set dihitung dari setiap sampel. Setelah itu, *variance* setiap komponen fitur dihitung dan digunakan untuk membuat pengkuantisasi untuk setiap komponen fitur yang menghasilkan vektor kuantisasi *step size* Q^u . Vektor kuantisasi *step size* Q^u ini digunakan untuk mengkuantisasi setiap vektor fitur tanda tangan dari proses *enrollment*. Yang terakhir adalah nilai rata-rata dari vektor-vektor fitur yang telah dikuantisasi ini digunakan sebagai template \bar{F}^u . Satu pasang (Q^u, \bar{F}^u) yaitu vektor kuantisasi *step size* dan template vektor fitur kemudian disimpan yang kemudian digunakan untuk memverifikasi tanda tangan *online* pengguna u yang diklaim. Detail rumus untuk menghasilkan vektor kuantisasi *step size* Q^u dan template vektor \bar{F}^u bisa dilihat dalam bab 2. Jadi setiap pengguna dalam setiap dataset akan diambil sepuluh tanda tangan asli yang sudah diekstrak menjadi histogram pada tahap sebelumnya untuk dijadikan template. Masing-masing pengguna akan menghasilkan dua buah file yaitu “quantizationstep_namauser.txt” dan “template_namauser.txt”.

Masing – masing template tanda tangan memiliki threshold minimum dan threshold maksimum. Threshold minimum dihitung dari rata-rata jarak antara tanda tangan latihan dengan template dikurangi dengan sepuluh kali standard deviasinya. Apabila hasilnya kurang dari nol, maka threshold minimum yang diambil adalah nol. Sedangkan threshold maksimum dihitung dari rata-rata jarak antara tanda tangan latihan dengan template ditambahkan dengan sepuluh kali standard deviasinya. Detail tentang bagaimana cara menghitung *threshold* minimum dan *threshold* maksimum dapat dilihat juga pada bab 2. Kemudian selanjutnya adalah membuat daftar 50 threshold yang merupakan nilai rentang dari threshold minimum sampai ke threshold maksimum. Masing-masing threshold akan dihitung *Confusion Matrix* nya.

Saat verifikasi berlangsung, tanda tangan uji baik tanda tangan asli, tanda tangan palsu terlatih, maupun tanda tangan palsu acak, yang mengklaim sebuah identitas u akan direpresentasikan kedalam fiturset histogram. Selanjutnya, vektor kuantisasi *step size* Q^u yang sudah disimpan dalam “quantizationstep_namauser.txt” digunakan untuk menghasilkan vektor fitur yang terkuantisasi dari tanda tangan masukan. Setelah itu, sistem menghitung jarak vektor yang terkuantisasi ini dengan template vektor yang sudah disimpan (\tilde{F}^u) dalam file “template_namauser.txt”. Jarak ini dihitung dengan menggunakan DTW dan juga dihitung dengan menggunakan *Manhattan Distance*. Tanda tangan diterima sebagai asli apabila jarak ketidaksamaan antara dua vektor ini lebih kecil dari *threshold* dan ditolak atau dianggap palsu apabila lebih besar dari *threshold*.

3.6. Skenario Uji Coba

Untuk menjawab rumusan masalah dan mencapai tujuan yang sudah ditetapkan di bab satu, maka dirancang skenario uji coba. Ada 3 skenario yang masing-masing akan membantu memperoleh jawaban dari rumusan masalah. Masing-masing dataset akan dilakukan uji coba yang sama secara bergantian. Ada dua jenis pemalsuan yaitu pemalsuan terlatih dan pemalsuan acak. Dalam pemalsuan terlatih, data uji yang digunakan adalah tanda tangan palsu yang diambil dari pemalsu yang sudah berlatih menirukan tanda tangan asli dengan cara diperlihatkan gambar tanda tangan asli tersebut. Kinerja sistem dapat diuji atau dibandingkan dengan menghitung nilai AUC (Area Under Curve). Sistem verifikasi yang nilai AUC nya mendekati satu maka sistem tersebut dapat memverifikasi dengan baik sedangkan sistem yang mendekati nilai 0.5 maka sistem tersebut sistem verifikasi yang kurang baik.

Dalam dataset SVC2004 task 1, masing-masing pengguna dari 40 pengguna memiliki 20 tanda tangan asli dan 20 tanda tangan palsu terlatih. Dari 20 tanda tangan asli yang ada, 10 tanda tangan akan diambil secara acak untuk membuat template tanda tangan untuk masing-masing pengguna. Sepuluh tanda tangan asli yang lain akan digunakan untuk melakukan uji coba sistem terhadap tanda tangan asli. Apabila sistem menerima tanda tangan tersebut maka hasil uji tersebut tergolong dalam True Positive sebaliknya apabila sistem menolak tanda tangan

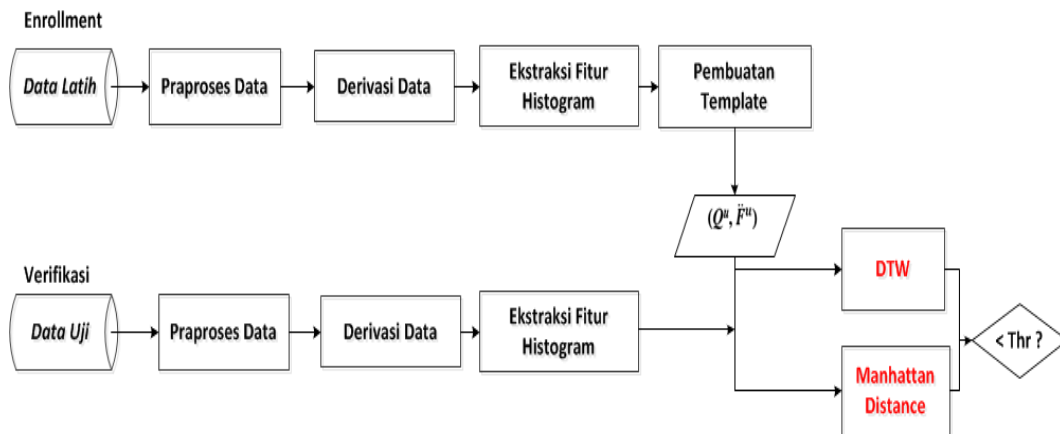
tersebut maka hasil uji tersebut tergolong dalam False Negative. Kemudian, tanda tangan palsu terdiri dari 20 tanda tangan palsu terlatih dan tanda tangan palsu acak yang diambil dari 39 pengguna yang lain ($39 \times 40 = 1560$) digunakan untuk melakukan uji coba sistem terhadap tanda tangan palsu. Apabila sistem menerima tanda tangan tersebut maka hasil uji tergolong dalam False Positive dan sebaliknya apabila sistem menolak tanda tangan tersebut maka hasil uji tergolong dalam True Negative. Oleh karena ada dua jenis tanda tangan palsu, maka ada dua jenis False Positive dan True Negative yaitu False Positive Skilled Forgery dan True Negative Skilled Forgery serta False Positive Random Forgery dan True Negative Random Forgery. Total data uji tanda tangan asli semua pengguna adalah $10 \times 40 = 400$. Total data uji tanda tangan palsu terlatih semua pengguna adalah $20 \times 40 = 800$. Total data uji tanda tangan palsu acak semua pengguna adalah $1560 \times 40 = 62400$.

Dalam dataset privat, masing-masing dari 19 pengguna memiliki 20 tanda tangan asli, dan 10 tanda tangan palsu. Sepuluh tanda tangan asli dari 20 tanda tangan yang ada, akan digunakan untuk membuat *template* tanda tangan untuk setiap pengguna. Sedangkan 10 tanda tangan sisanya digunakan untuk menghitung menguji apakah True Positive atau False Negative. Selain itu, 10 tanda tangan palsu yang terlatih digunakan untuk menguji apakah True Negative Skilled Forgery atau False Positive Skilled Forgery. Sedangkan tanda tangan selain dari tanda tangan pengguna yang bersangkutan akan digunakan untuk menguji apakah True Negative Random Forgery atau False Positive Random Forgery ($18 \times 30 = 540$). Total data uji tanda tangan asli semua pengguna adalah $10 \times 19 = 190$. Total data uji tanda tangan palsu terlatih semua adalah $10 \times 19 = 190$. Total data uji tanda tangan palsu acak semua pengguna adalah $540 \times 19 = 10260$.

3.6.1 Uji Coba Perbandingan *Dynamic Time Warping* dan *Manhattan Distance*

Salah satu perbedaan penelitian ini dan penelitian sebelumnya adalah cara mengukur jarak antara dua buah vektor yang dibentuk dari fitur-fitur histogram, yaitu vektor tanda tangan yang diklaim oleh pengguna dengan vektor *template* yang sudah disimpan dengan 10 data latih dari pengguna tersebut. Seiring dengan latar belakang dan tujuan penelitian bahwa dengan menggunakan *Dynamic Time*

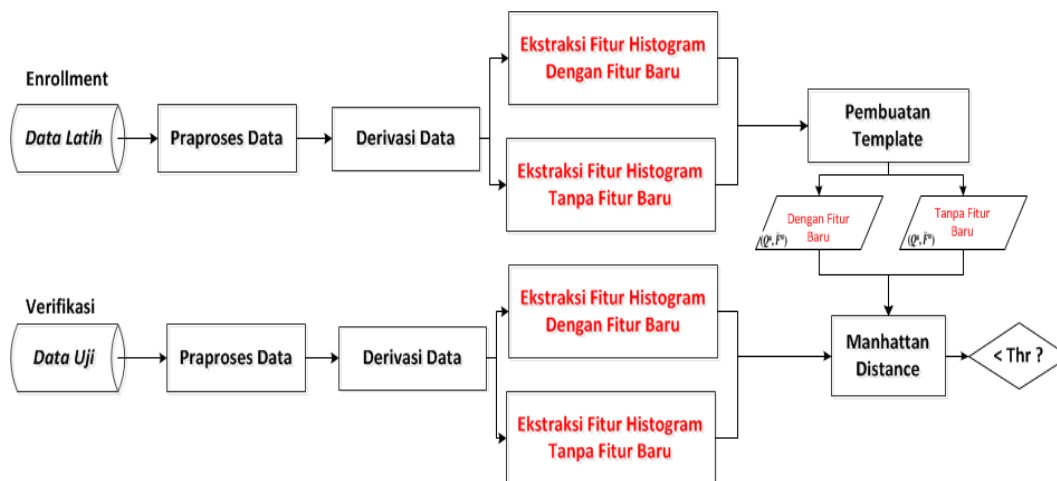
Warping diharapkan akan meningkatkan kinerja dibandingkan dengan sistem yang menggunakan *Manhattan Distance* sebagai pengukur jarak. Untuk membandingkan kedua metode tersebut maka, variabel-variabel yang lain disamakan termasuk fitur-fitur histogram yang digunakan. Untuk ujicoba kali ini, fitur histogram yang digunakan adalah fitur histogram yang asli tanpa ada penambahan fitur yang baru.



Gambar 3.6. Proses Verifikasi Skenario Perbandingan DTW dan Manhattan

3.6.2. Uji Coba Perbandingan Kinerja Dengan dan Tanpa Penambahan Fitur Baru

Dalam skenario ini akan dibandingkan kinerja sistem yang dibangun tanpa menggunakan fitur-fitur baru dengan sistem yang dibangun dengan menggunakan fitur-fitur yang baru yang diturunkan dari histogram X dan Y. Untuk melakukan pengujian ini, variabel-variabel yang lain akan disamakan termasuk metode pengukuran jarak yang digunakan. Dalam skenario ini, pengukuran jarak yang akan digunakan adalah pengukuran jarak yang asli yaitu *Manhattan Distance*.



Gambar 3.7. Proses Verifikasi Skenario Perbandingan Dengan dan Tanpa Fitur Baru

3.6.4 Uji Coba Peringkat Fitur

Dalam skenario ini akan dilakukan pemeringkatan seluruh fitur yang digunakan dalam dataset Privat dengan menggunakan tiga metode yaitu Gain Ratio, Correlation, dan RelieF. Hasil peringkat dari 16 fitur-fitur baru dicatat dalam subbab ini.

3.7. Analisis Hasil

Berdasarkan hasil dari uji coba yang dilakukan pada setiap skenario, akan dilakukan analisis untuk mengetahui, yang pertama tentang bagaimana peran fitur-fitur histogram yang baru yaitu X dan Y terhadap performa verifikasi tanda tangan *online*. Yang kedua untuk mengetahui bagaimana peran DTW terhadap hasil verifikasi tanda tangan dinamis apabila dibandingkan dengan *Manhattan Distance*. Yang ketiga adalah untuk mengetahui bagaimana peringkat fitur terhadap hasil kelas dengan menggunakan metode.

3.8. Penyusunan Kesimpulan dan Saran Pengembangan Penelitian

Setelah seluruh rangkaian penelitian sudah dilakukan, kemudian kesimpulan terhadap hasil penelitian disusun. Kesimpulan yang dibuat akan menjawab rumusan masalah dan tujuan yang sudah ditentukan saat awal pembuatan penelitian di bab pendahuluan. Selain itu saran mengenai penelitian selanjutnya juga disusun pada tahap ini.

BAB 4

HASIL DAN PEMBAHASAN

Bab ini menjelaskan mengenai proses uji coba sistem verifikasi tanda tangan online berdasarkan skenario uji coba yang sudah dibuat. Hal-hal yang berkaitan dengan proses uji coba yang akan dijelaskan dalam bab ini antara lain data uji coba, lingkungan uji coba, skenario uji coba, pelaksanaan dan hasil uji coba, serta analisis hasil uji coba.

4.1. Data Uji Coba

Data yang digunakan dalam uji coba kali ini ada 3 dataset antara lain dataset SVC2004 dan dataset Privat. Penjelasan detail mengenai bagaimana sejarah dataset SVC2004 dapat dilihat pada bab II. Sedangkan penjelasan tentang bagaimana cara untuk mendapatkan dataset publik tersebut dapat dilihat pada bab III, dan juga sekaligus penjelasan mengenai bagaimana peneliti mengumpulkan dataset privat. Sedangkan pada bab ini akan dijelaskan mengenai dataset SVC2004 dan Privat yang akan digunakan dalam uji coba.

Dataset SVC2004 yang digunakan adalah dataset task1 yang sesuai dengan kondisi perangkat mobile dimana tidak ada informasi tekanan dan orientasi pen. Dataset tersebut berisi tanda tangan dari 40 user. Masing-masing user disediakan 20 tanda tangan asli, dan 20 tanda tangan palsu dengan pemalsuan terlatih. Dataset privat yang diambil adalah dataset dari 19 orang. Format data yang diambil sama yaitu posisi x, posisi y, *timestamp* dan *button status*. Masing-masing orang diambil 20 tanda tangan asli dan 10 tanda tangan palsu yang terlatih.

Tabel 4.1 Jumlah User Per Dataset

Dataset	Jumlah User	Asli	Palsu terlatih
SVC2004	40	20/user	20/user
PRIVAT	19	20/user	10/user

4.2. Perangkat Uji Coba

Berikut adalah spesifikasi perangkat mobile yang digunakan untuk mengambil tanda tangan uji Privat:

Tabel 4.2 Spesifikasi Perangkat Mobile

Fitur	Spesifikasi
Merk	Coolpad R18
OS	Android v 5.1.1
CPU	Qualcomm Tech., Inc., MSM8929 Octa-Core 1.4 Ghz
RAM	3GB
Resolution	1280 x 720
Browser	Mozilla For Android 60.0.2

Uji coba dilakukan di dalam lingkungan virtualisasi dengan menggunakan *Oracle VM Virtual Box* versi 5.1.26. Virtualisasi dilakukan di dalam sebuah notebook dengan spesifikasi sebagai berikut:

Tabel 4.3 Spesifikasi Perangkat Keras Notebook Host

Perangkat	Spesifikasi
Processor	Intel(R) Core(TM) i3-2350M CPU @ 2.30Ghz 2.30 Ghz
RAM	8 GB
System Type	64-bit Operating System (Windows 7)

Sedangkan spesifikasi mesin *guest* di dalam *Virtual Box* antara lain sebagai berikut:

Tabel 4.4 Spesifikasi Perangkat Keras Mesin Virtualisasi

Jenis	Spesifikasi
Processor	1 Virtual CPU
RAM	1024 MB
System Type	64-bit Operating System (Ubuntu Server 14.04)

Sistem verifikasi tanda tangan dinamis dikembangkan dengan menggunakan bahasa pemrograman *python* dengan framework dan beberapa library antara lain sebagai berikut:

Tabel 4.5 Spesifikasi Perangkat Lunak

Framework/Lib	Versi
Python	Python 2.7.6
Django Framework	1.11.1
Numpy Library	1.12.1
Pandas Library	0.20.1
Matplotlib Library	2.0.2
FastDtw Libray	0.3.2

Untuk melakukan pemeringkatan fitur, digunakan tools Weka dengan spesifikasi sebagai berikut :

Tabel 4.6 Spesifikasi Tools Pemeringkatan Fitur

Tools	Versi
WEKA	3.8.1
Java	1.7

4.3. Pelaksanaan dan Hasil Uji Coba

Pada bagian ini akan dijelaskan mengenai detail hasil uji coba yang dilakukan berdasarkan skenario yang telah dibuat pada sub bab sebelumnya. Sebagai catatan, dalam penulisan hasil digunakan singkatan antara lain FP adalah singkatan dari False Positive, FN singkatan dari False Negatif, TN singkatan dari True Negative, dan FP singkatan dari False Positive. Sedangkan SF singkatan dari Skilled Forgery, RF singkatan dari Random Forgery, Thr adalah singkatan dari Threshold, PrecisionSF adalah nilai Precision untuk kasus Skilled Forgery, PrecisionRF adalah nilai Precision untuk kasus Random Forgery. Dalam sistem klasifikasi biner atau klasifikasi nilai Recall dan Precision saling dbertolak belakang. Apabaila nilai Recall tinggi maka nilai Precision akan turun, dan apabila nilai Recall besar maka nilai Precision akan turun. Oleh karena itu dihitung nilai Recall dan Precision di 50 nilai threshold yang berbeda, dan kemudian dicari titik temu dari kurva Recall dan Precision. Threshold di titik temu tersebut merupakan titik optimal dimana sistem verifikasi dapat dibandingkan. Selain menggunakan recall, precision, accuracy, F-measure, digunakan juga nilai AUC.

Yang dicatat dalam sub-bab ini antara lain confusion matrix di 25 threshold pertama untuk kemudahan membaca, sedangkan tabel confusion matrix yang lengkap di 50 threshold dapat dilihat pada lampiran. Setelah itu diberikan grafik ROC untuk mencari titik threshold optimal yang merupakan titik temu antara Recall dan Precision baik PrecisionSF (Precision Skilled Forgery) maupun PrecisionRF (Precision Random-Forgery). Pada titik threshold optimal tersebut kemudian dicari tabel confusion matrixnya dan kemudian dihitung nilai recall, precision, F-measure, dan accuracy. Setelah itu, dihitung nilai AUC nya.

4.3.1 Perbandingan *Dynamic Time Warping* dan *Manhattan Distance*

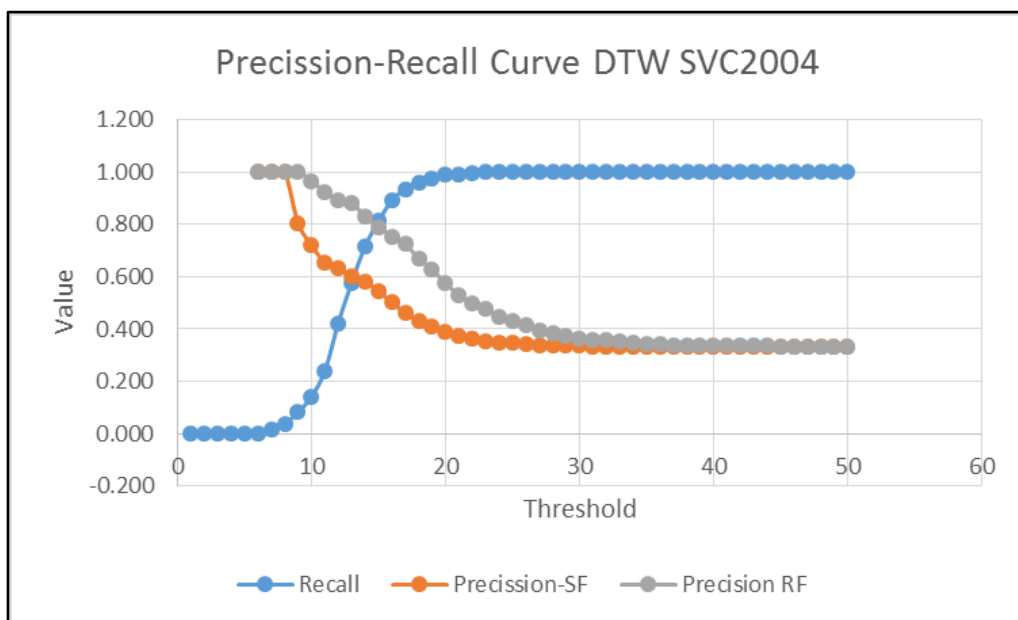
a Dataset SVC2004

Berikut adalah hasil dari confusion matrix di 25 threshold pertama metode DTW pada dataset SVC2004. Tabel lengkap confusion matrix dapat dilihat pada lampiran A.1.1.

Tabel 4.7 Confusion Matrix DTW pada dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	400	800	0	800	0
2	0	400	800	0	800	0
3	0	400	800	0	800	0
4	0	400	800	0	800	0
5	0	400	800	0	800	0
6	1	399	800	0	800	0
7	7	393	800	0	800	0
8	15	385	800	0	800	0
9	33	367	792	8	800	0
10	57	343	778	22	798	2
11	96	304	749	51	792	8
12	169	231	701	99	779	21
13	231	169	648	152	769	31
14	286	114	595	205	741	59
15	325	75	528	272	713	87
16	356	44	448	352	684	116
17	374	26	361	439	658	142
18	384	16	294	506	610	190
19	390	10	236	564	566	234
20	395	5	185	615	511	289
21	395	5	137	663	446	354
22	398	2	97	703	402	398
23	400	0	75	725	362	438
24	400	0	54	746	309	491
25	400	0	42	758	267	533

Kemudian, berikut adalah grafik Recall x PrecissionSF dan Recall x PrecissionRF pada keseluruhan threshold.

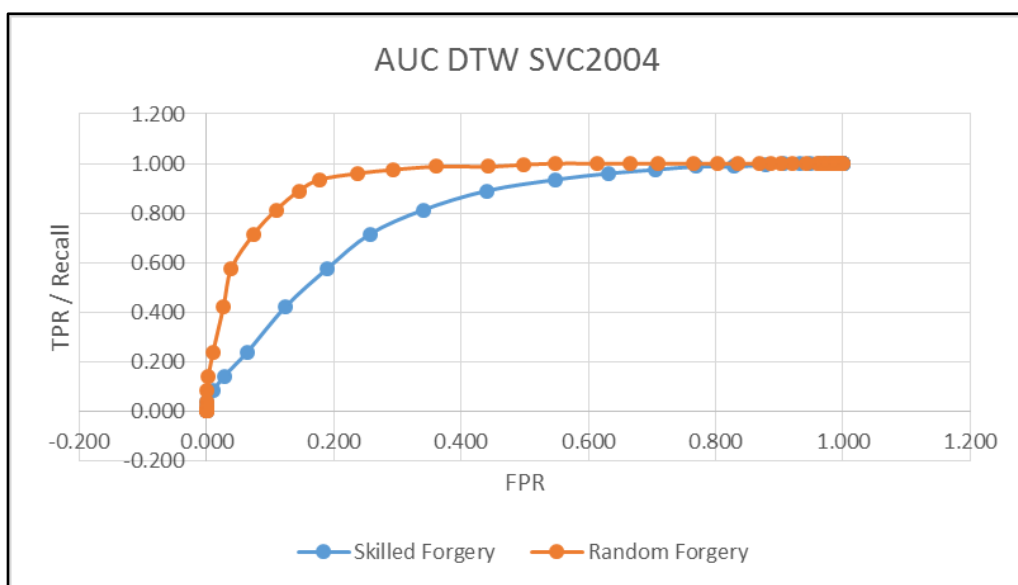


Gambar 4.1. Kurva Precision-Recall metode DTW dalam dataset SVC2004

Titik threshold optimal untuk kasus Skilled Forgery adalah 13.2990881 dan titik threshold optimal untuk kasus Random Forgery adalah 15.5709825. Berikut adalah hasil confusion matrix pada titik threshold tersebut.

Tabel 4.8 Confusion Matrix DTW threshold optimal di dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
13.2990881	250	150	631	169	759	41
15.5709825	345	55	493	307	702	98



Gambar 4.2. AUC metode DTW dalam dataset SVC2004

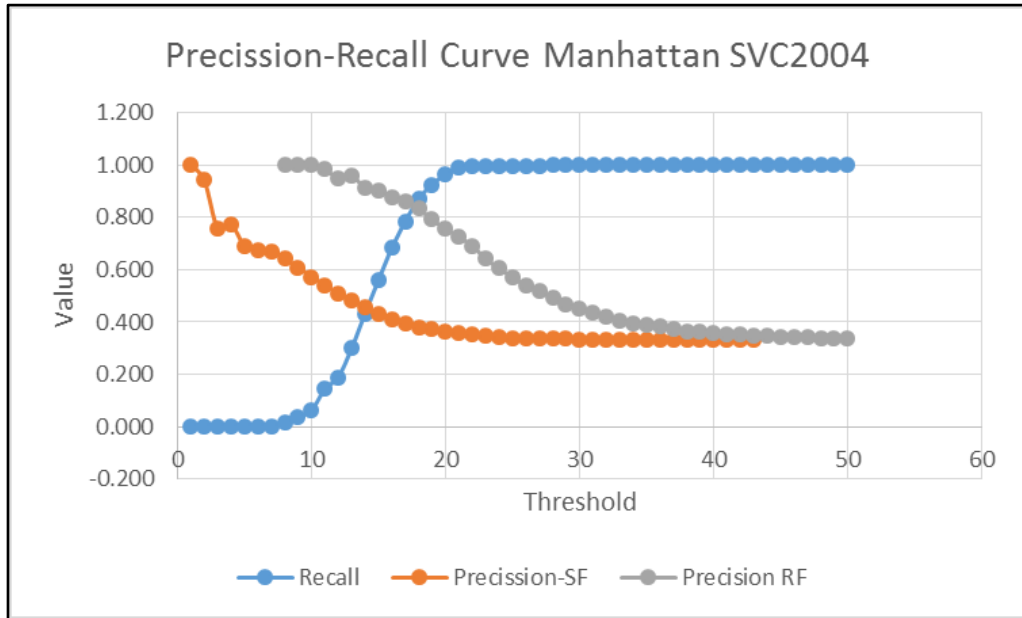
Nilai AUC untuk Skilled Forgery adalah 0.79568 dan nilai AUC untuk Random Forgery adalah 0.937864063.

Sedangkan berikut adalah tabel confusion matrix metode Manhattan di 25 threshold pertama pada dataset SVC2004. Tabel lengkap ada di lampiran A.1.2

Tabel 4.9 Confusion Matrix Manhattan pada dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	400	800	0	800	0
2	0	400	800	0	800	0
3	0	400	800	0	800	0
4	0	400	800	0	800	0
5	0	400	800	0	800	0
6	0	400	800	0	800	0
7	0	400	800	0	800	0
8	6	394	800	0	800	0
9	16	384	799	1	800	0
10	25	375	792	8	800	0
11	58	342	783	17	799	1
12	76	324	766	34	796	4
13	120	280	742	58	795	5
14	173	227	715	85	783	17
15	224	176	675	125	775	25
16	273	127	625	175	762	38
17	314	86	566	234	748	52
18	348	52	506	294	730	70
19	370	30	442	358	702	98
20	385	15	386	414	676	124
21	396	4	326	474	650	150
22	397	3	273	527	622	178
23	398	2	229	571	578	222
24	399	1	186	614	539	261
25	399	1	153	647	500	300

Kemudian, berikut adalah grafik Recall x PrecissionSF dan Recall x PrecissionRF pada keseluruhan threshold.

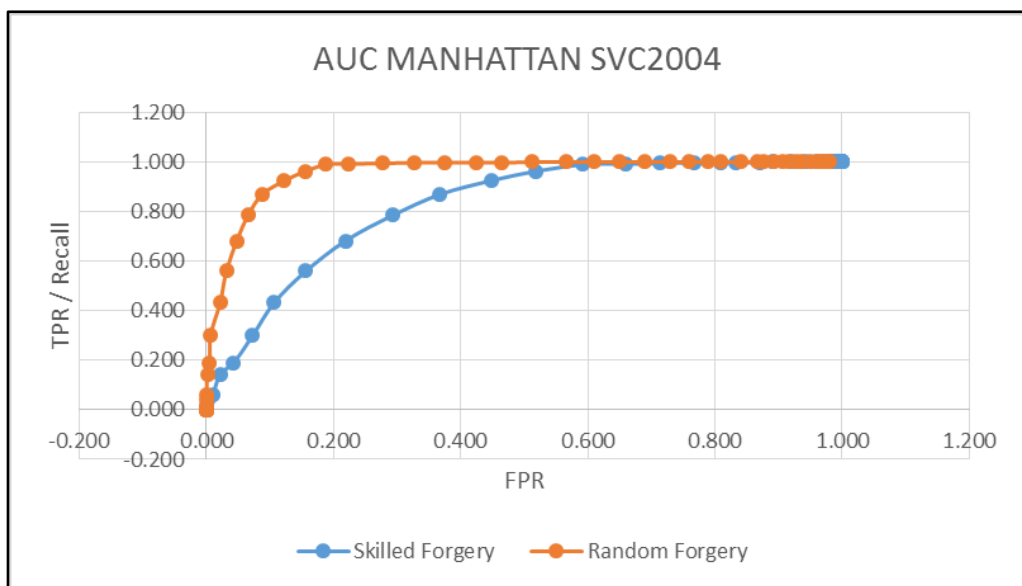


Gambar 4.3. Kurva Precision-Recall metode Manhattan dalam dataset SVC2004

Titik threshold optimal untuk kasus Skilled Forgery adalah 13.95533227 dan titik threshold optimal untuk kasus Random Forgery adalah 18.55578408. Berikut adalah hasil confusion matrix pada titik threshold tersebut.

Tabel 4.10 Confusion Matrix Manhattan threshold optimal di dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
13.95533227	171	229	716	84	784	16
18.55578408	364	36	472	328	713	87



Gambar 4.4. AUC metode Manhattan dalam dataset SVC2004

Nilai AUC untuk Skilled Forgery adalah 0.821909375 dan nilai AUC untuk Random Forgery adalah 0.93815

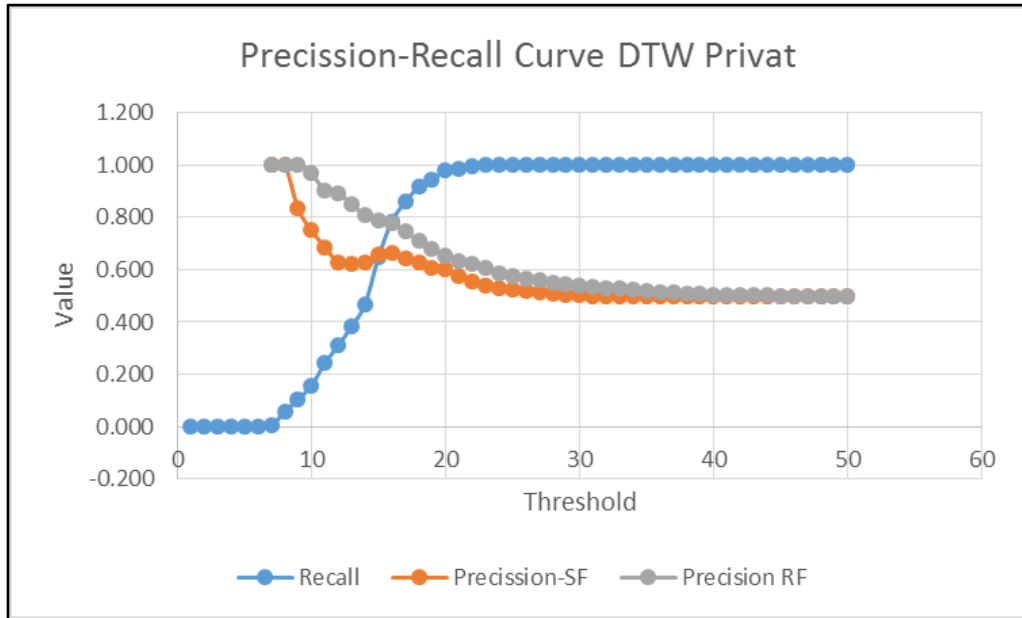
b Dataset Privat

Berikut adalah hasil dari confusion matrix di 25 threshold pertama metode DTW pada dataset Privat. Tabel lengkap confusion matrix dapat dilihat pada lampiran A.1.3.

Tabel 4.11 Confusion Matrix DTW pada dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	190	190	0	190	0
2	0	190	190	0	190	0
3	0	190	190	0	190	0
4	0	190	190	0	190	0
5	0	190	190	0	190	0
6	0	190	190	0	190	0
7	1	189	190	0	190	0
8	11	179	190	0	190	0
9	20	170	186	4	190	0
10	30	160	180	10	189	1
11	46	144	169	21	185	5
12	59	131	155	35	183	7
13	73	117	146	44	177	13
14	89	101	137	53	169	21
15	123	67	126	64	157	33
16	149	41	114	76	147	43
17	163	27	99	91	134	56
18	174	16	87	103	119	71
19	179	11	74	116	105	85
20	186	4	66	124	92	98
21	187	3	53	137	81	109
22	189	1	38	152	76	114
23	190	0	28	162	67	123
24	190	0	21	169	57	133
25	190	0	16	174	51	139

Kemudian, berikut adalah grafik Recall x PrecissionSF dan Recall x PrecissionRF pada keseluruhan threshold.

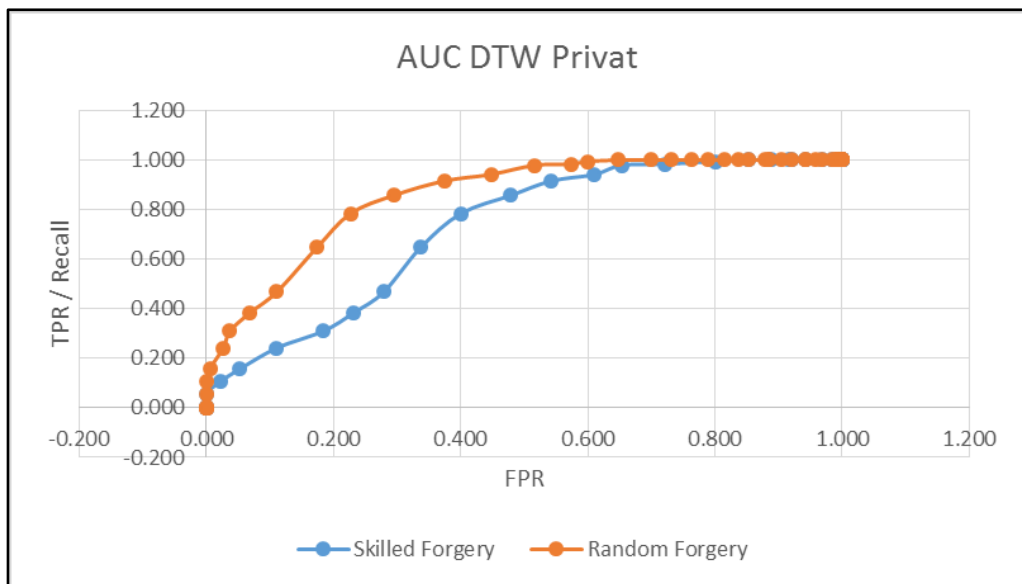


Gambar 4.5. Kurva Precision-Recall metode DTW dalam dataset Privat

Titik threshold optimal untuk kasus Skilled Forgery adalah 15.0417778 dan titik threshold optimal untuk kasus Random Forgery adalah 16.6650682. Berikut adalah hasil confusion matrix pada titik threshold tersebut.

Tabel 4.12 Confusion Matrix DTW threshold optimal di dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
15.0417778	123	67	126	64	161	29
16.6650682	161	29	102	88	141	49



Gambar 4.6. AUC metode DTW dalam dataset Privat

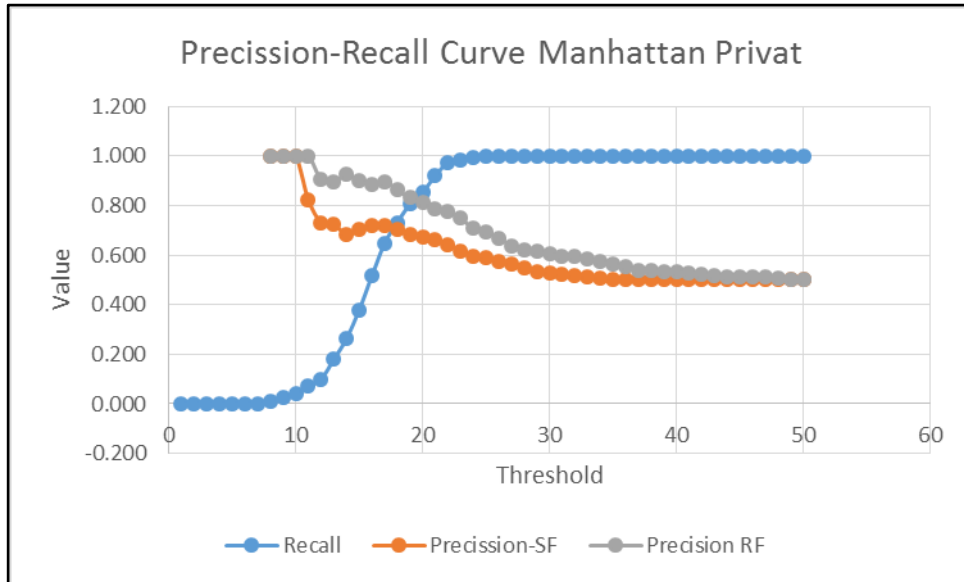
Nilai AUC untuk Skilled Forgery adalah 0.721939058 dan nilai AUC untuk Random Forgery adalah 0.852742382

Sedangkan berikut adalah tabel confusion matrix metode Manhattan di 25 threshold pertama pada dataset Privat. Tabel lengkap ada di lampiran A.1.4

Tabel 4.13 Confusion Matrix Manhattan pada dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	190	190	0	190	0
2	0	190	190	0	190	0
3	0	190	190	0	190	0
4	0	190	190	0	190	0
5	0	190	190	0	190	0
6	0	190	190	0	190	0
7	0	190	190	0	190	0
8	2	188	190	0	190	0
9	5	185	190	0	190	0
10	8	182	190	0	190	0
11	14	176	187	3	190	0
12	19	171	183	7	188	2
13	34	156	177	13	186	4
14	50	140	167	23	186	4
15	72	118	160	30	182	8
16	98	92	152	38	177	13
17	123	67	142	48	176	14
18	139	51	132	58	168	22
19	153	37	119	71	159	31
20	162	28	111	79	153	37
21	175	15	100	90	142	48
22	185	5	86	104	136	54
23	187	3	74	116	127	63
24	189	1	62	128	112	78
25	190	0	57	133	106	84

Kemudian, berikut adalah grafik Recall x PrecissionSF dan Recall x PrecissionRF pada keseluruhan threshold.

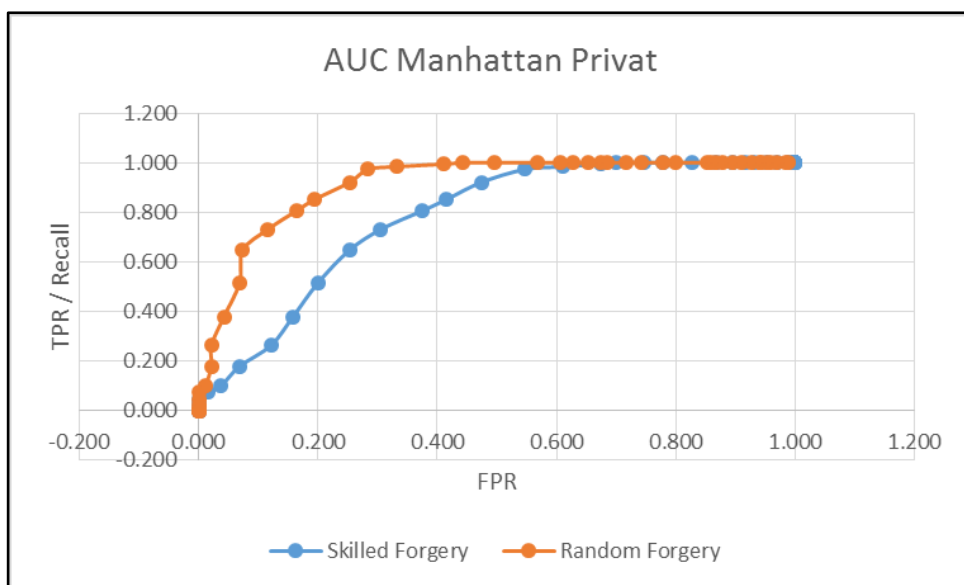


Gambar 4.7. Kurva Precision-Recall metode Manhattan dalam dataset Privat

Titik threshold optimal untuk kasus Skilled Forgery adalah 18.3509 dan titik threshold optimal untuk kasus Random Forgery adalah 20.2178. Berikut adalah hasil confusion matrix pada titik threshold tersebut.

Tabel 4.14 Confusion Matrix Manhattan threshold optimal di dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
18.3509	145	45	129	61	169	21
20.2178	164	26	108	82	151	39



Gambar 4.8. AUC metode Manhattan dalam dataset Privat

Nilai AUC untuk Skilled Forgery adalah 0.775540166 dan nilai AUC untuk Random Forgery adalah 0.900193906

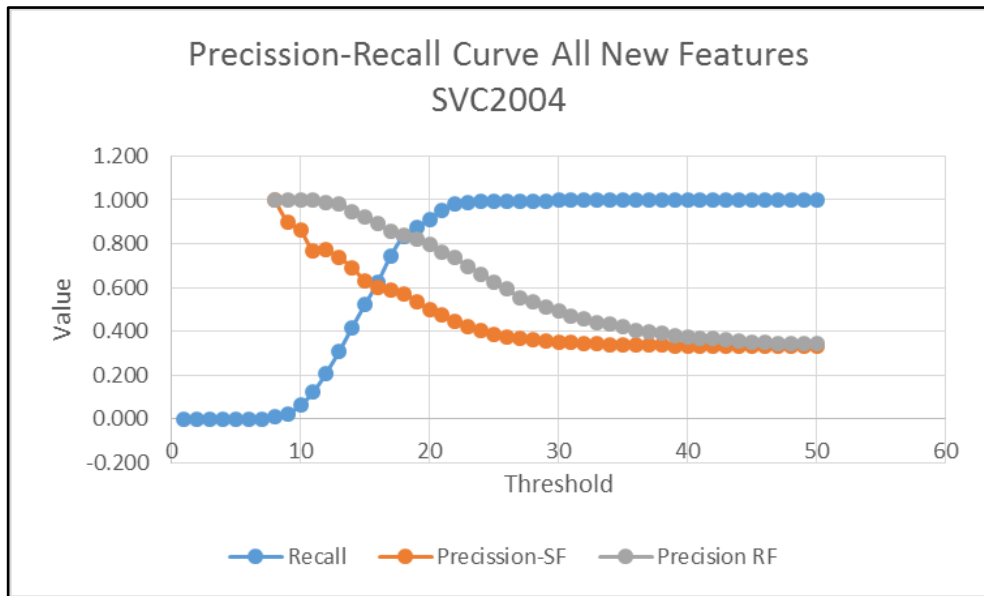
4.3.2 Perbandingan Kinerja Fitur Baru

a Dataset SVC2004

Karena hasil skenario tanpa penggunaan fitur baru ekuivalen dengan skenario Manhattan Distance maka hasil dapat dilihat pada tabel 4.9, gambar 4.3, tabel 4.10 dan gambar 4.4. Sedangkan berikut adalah confusion matrix hasil sistem dengan menggunakan fitur baru dalam dataset SVC2004 di 25 threshold pertama. Tabel lengkap ada di lampiran A.2.1

Tabel 4.15 Confusion Matrix Dengan Fitur Baru pada dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	400	800	0	800	0
2	0	400	800	0	800	0
3	0	400	800	0	800	0
4	0	400	800	0	800	0
5	0	400	800	0	800	0
6	0	400	800	0	800	0
7	0	400	800	0	800	0
8	5	395	800	0	800	0
9	9	391	799	1	800	0
10	25	375	796	4	800	0
11	49	351	785	15	800	0
12	84	316	775	25	799	1
13	123	277	756	44	798	2
14	167	233	726	74	791	9
15	209	191	678	122	782	18
16	250	150	635	165	769	31
17	298	102	592	208	751	49
18	334	66	551	249	736	64
19	350	50	499	301	724	76
20	364	36	434	366	707	93
21	382	18	374	426	682	118
22	392	8	314	486	659	141
23	396	4	253	547	628	172
24	397	3	211	589	598	202
25	397	3	172	628	562	238

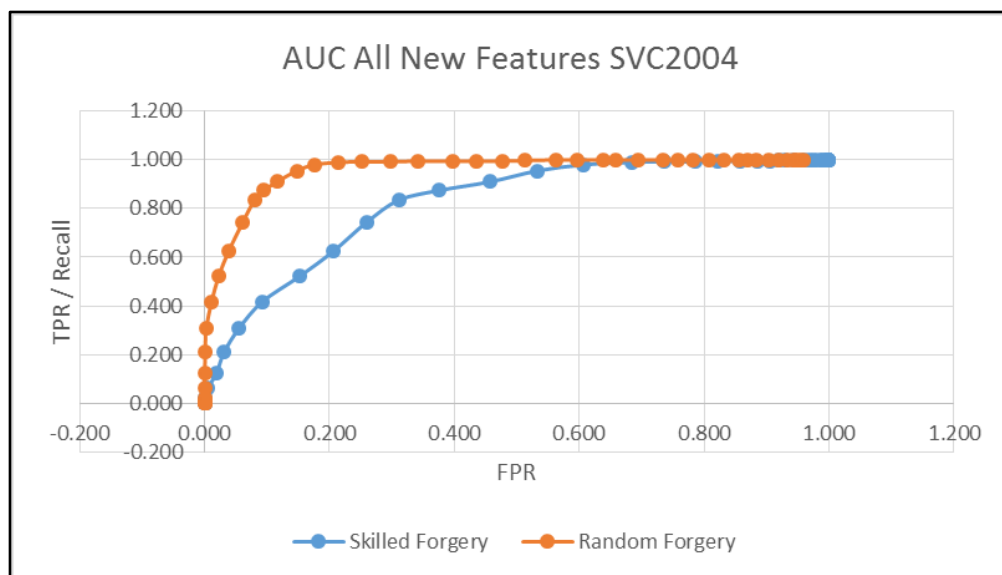


Gambar 4.9. Kurva Precision-Recall Semua Fitur Baru dalam dataset SVC2004

Titik threshold optimal untuk kasus Skilled Forgery adalah 16.1236193 dan titik threshold optimal untuk kasus Random Forgery adalah 19.2049799. Berikut adalah hasil confusion matrix pada titik threshold tersebut.

Tabel 4.16 Confusion Matrix Fitur Baru Threshold Optimal dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
16.1236193	253	147	632	168	768	32
19.2049799	355	45	487	313	714	86



Gambar 4.10. AUC Semua Fitur Baru Dataset SVC2004

Nilai AUC untuk Skilled Forgery adalah 0.820025 dan nilai AUC untuk Random Forgery adalah 0.917389063. Selanjutnya adalah hasil peringkat fitur berdasarkan nilai AUC-SF. Pemeringkatan diurutkan berdasarkan nilai AUC-SF dari yang paling besar ke yang paling kecil.

Tabel 4.17 Peringkat Fitur Berdasarkan Nilai AUC_SF Dataset SVC2004

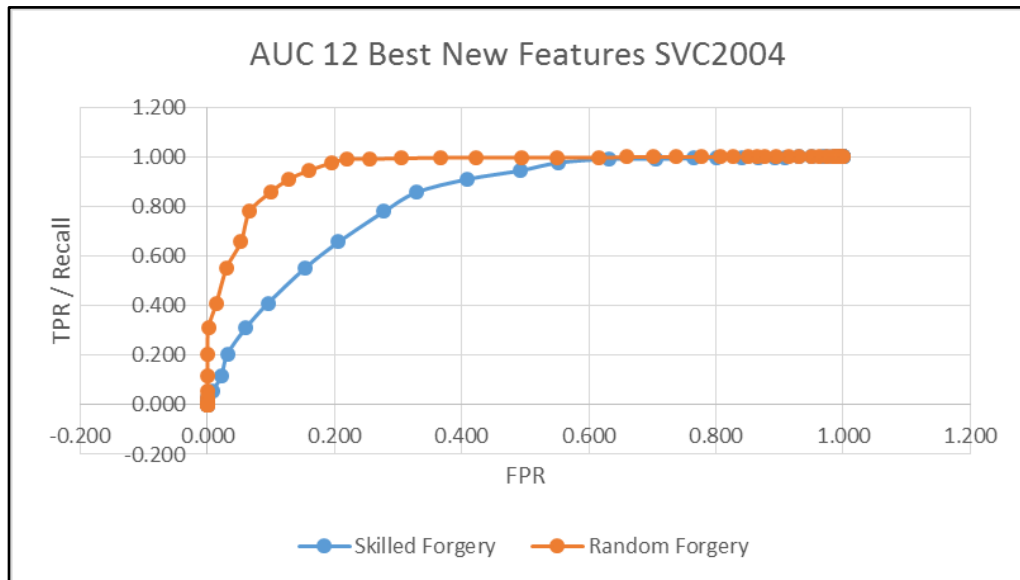
No	Nama Fitur	auc_sf
1	x_4	0.824095
2	x_3	0.82272
3	x_1	0.822467
4	x_2	0.822434
5	y_6	0.821923
6	x_8	0.821891
7	x_7	0.821564
8	y_8	0.821314
9	y_7	0.821045
10	y_1	0.820977
11	y_3	0.820959
12	y_2	0.820847
13	y_4	0.820805
14	x_5	0.820105
15	x_6	0.81887
16	y_5	0.815997

Berikut adalah hasil kinerja AUC yang didapatkan dari 4 besar, 8 besar, 12 besar, dan semua fitur baru

Tabel 4.18 Nilai AUC_SF dan AUC_RF per kelompok kelipatan 4 serta 13,14, dan 15 besar Dataset SVC2004

Fitur	AUC_SF	AUC_RF
4 besar (x_4,x_3,x_1,x_2)	0.826469	0.949013
8 besar (x_4,x_3,x_1,x_2,y_6,x_8, x_7,y_8)	0.82743	0.953858
12 besar (x_4,x_3,x_1,x_2,y_6,x_8, x_7,y_8,y_7,y_1,y_3,y_2)	0.828453	0.9551
13 besar	0.82501	0.95291
14 besar	0.82407	0.95111
15 besar.	0.82136	0.94837
16 (Semua fitur baru)	0.820025	0.917389063

Dapat dilihat bahwa nilai kinerja terbaik didapatkan pada saat menggunakan 12 besar dari peringkat fitur yaitu dengan AUC-SF 0.828453 dan 0.9551. Berikut adalah confusion matrix dari penggunaan 12 fitur baru tersebut di 25 threshold pertama

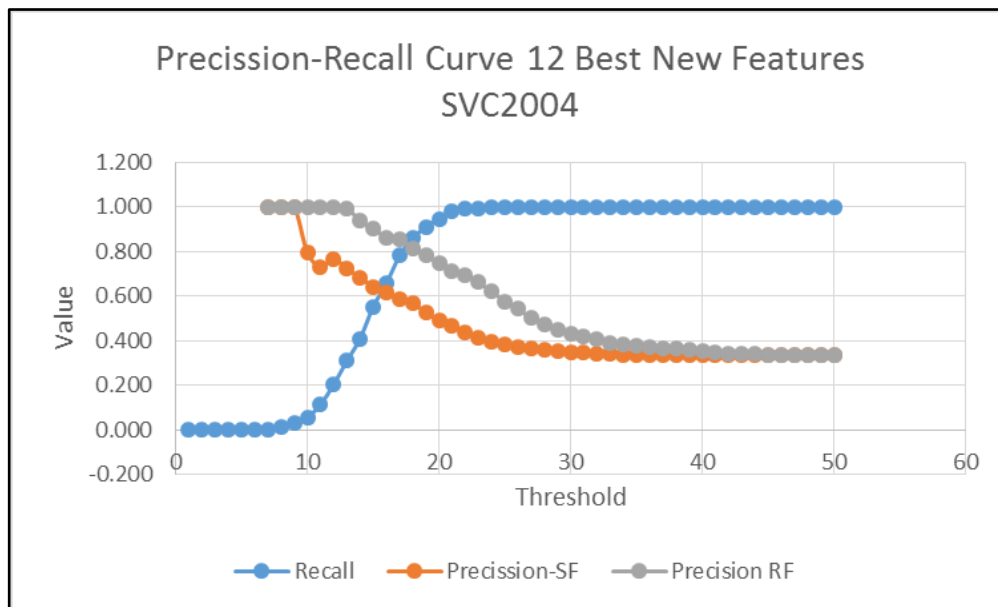


Gambar 4.11. AUC Dengan 12 Fitur Baru peringkat Tertinggi Dataset SVC2004

Tabel 4.19 Confusion Matrix Dengan 12 Fitur Peringkat Tetinggi Dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	400	800	0	800	0
2	0	400	800	0	800	0
3	0	400	800	0	800	0
4	0	400	800	0	800	0
5	0	400	800	0	800	0
6	0	400	800	0	800	0
7	1	399	800	0	800	0
8	5	395	800	0	800	0
9	13	387	800	0	800	0
10	23	377	794	6	800	0
11	46	354	783	17	800	0
12	82	318	775	25	800	0
13	124	276	752	48	799	1
14	164	236	724	76	789	11
15	221	179	677	123	776	24
16	263	137	635	165	758	42

17	312	88	578	222	747	53
18	343	57	537	263	721	79
19	364	36	474	326	698	102
20	378	22	407	393	673	127
21	391	9	358	442	643	157
22	397	3	294	506	624	176
23	397	3	235	565	596	204
24	398	2	188	612	556	244
25	399	1	159	641	507	293



Gambar 4.12. ROC Dengan 12 Fitur Baru peringkat Tertinggi Dataset SVC2004

Titik threshold optimal untuk kasus Skilled Forgery adalah 15.90612845 dan titik threshold optimal untuk kasus Random Forgery adalah 18.66426896. Berikut adalah hasil confusion matrix pada titik threshold tersebut.

Tabel 4.20 Confusion Matrix 12 Fitur Baru Threshold Optimal dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
15.90613	259	141	641	159	760	40
18.66427	356	44	486	314	702	98

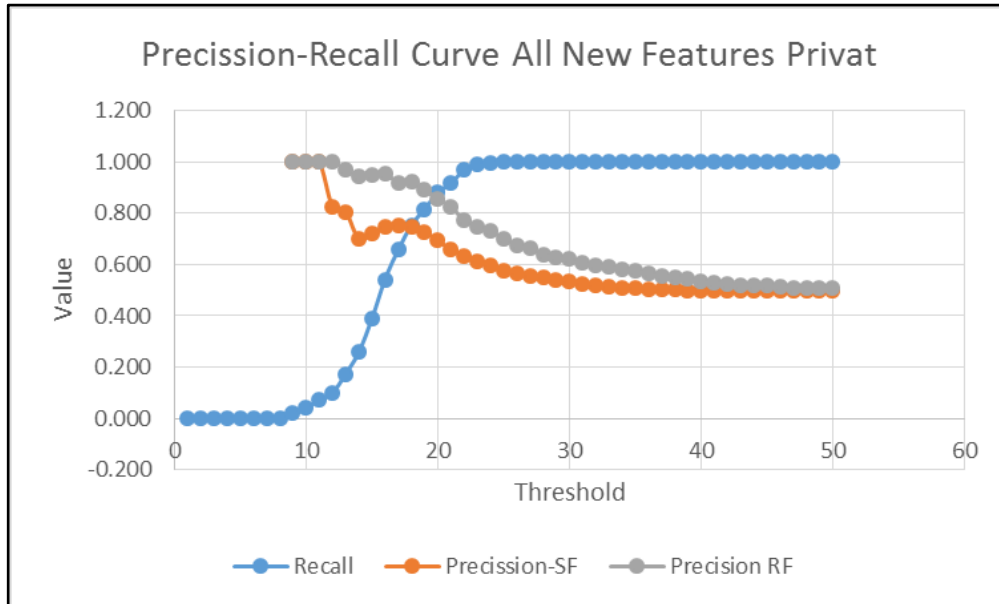
b Dataset Privat

Karena hasil skenario tanpa penggunaan fitur baru ekuivalen dengan skenario Manhattan Distance maka hasil dapat dilihat pada tabel 4.13, gambar 4.7, tabel 4.14 dan gambar 4.8. Sedangkan berikut adalah confusion matrix hasil sistem dengan

menggunakan seluruh fitur baru dalam dataset Privat di 25 threshold pertama. Tabel lengkap ada di lampiran A.2.2.

Tabel 4.21 Confusion Matrix Dengan Fitur Baru pada dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	190	190	0	190	0
2	0	190	190	0	190	0
3	0	190	190	0	190	0
4	0	190	190	0	190	0
5	0	190	190	0	190	0
6	0	190	190	0	190	0
7	0	190	190	0	190	0
8	0	190	190	0	190	0
9	4	186	190	0	190	0
10	8	182	190	0	190	0
11	14	176	190	0	190	0
12	19	171	186	4	190	0
13	33	157	182	8	189	1
14	49	141	169	21	187	3
15	74	116	161	29	186	4
16	103	87	155	35	185	5
17	125	65	149	41	179	11
18	143	47	141	49	178	12
19	155	35	131	59	171	19
20	167	23	117	73	162	28
21	174	16	99	91	153	37
22	184	6	84	106	136	54
23	188	2	72	118	127	63
24	189	1	63	127	120	70
25	190	0	51	139	109	81

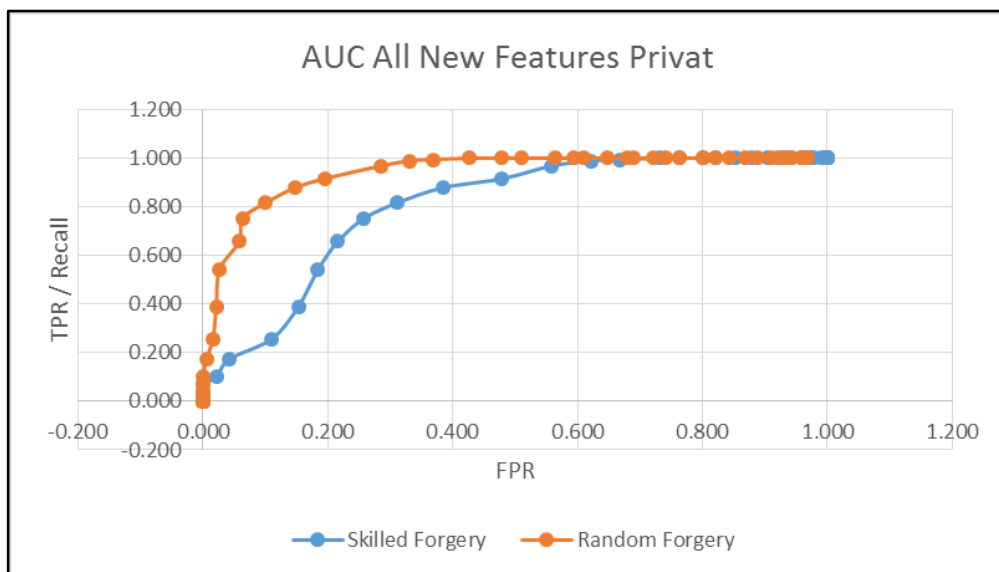


Gambar 4.13. Kurva Precision-Recall Semua Fitur Baru dalam dataset Privat

Titik threshold optimal untuk kasus Skilled Forgery adalah 18.5106252 dan titik threshold optimal untuk kasus Random Forgery adalah 20.5276371. Berikut adalah hasil confusion matrix pada titik threshold tersebut.

Tabel 4.22 Confusion Matrix Fitur Baru Threshold Optimal dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
18.5106252	149	41	134	56	167	23
20.5276371	171	19	111	79	152	38



Gambar 4.14. AUC Dengan Fitur Baru Dataset Privat

Nilai AUC untuk Skilled Forgery adalah 0.799792244 dan nilai AUC untuk Random Forgery adalah 0.908407202.

Berikut adalah hasil peringkat fitur berdasarkan nilai AUC-SF. Pemeringkatan diurutkan berdasarkan nilai AUC-SF dari yang paling besar ke yang paling kecil.

Tabel 4.23 Peringkat Fitur Berdasarkan Nilai AUC_SF Dataset Privat

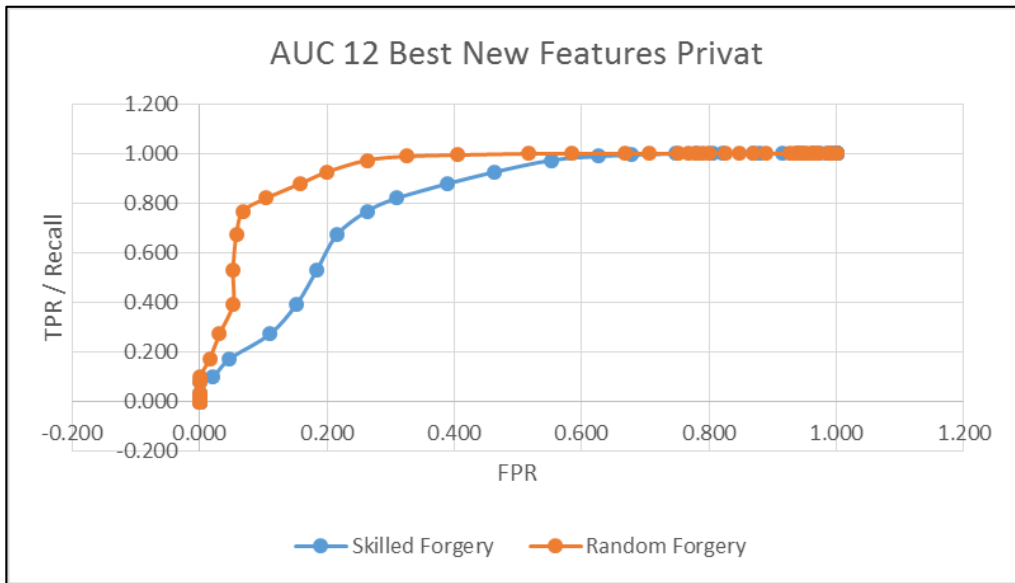
No	Nama Fitur	auc_sf
1	x_6	0.783006
2	x_5	0.782202
3	x_3	0.780997
4	x_4	0.780402
5	y_5	0.779391
6	y_4	0.7791
7	x_7	0.778504
8	y_7	0.778504
9	y_6	0.777202
10	y_8	0.777008
11	x_2	0.776759
12	y_2	0.774737
13	x_1	0.774058
14	y_1	0.774058
15	y_3	0.773864
16	x_8	0.773767

Tabel 4.24 Nilai AUC_SF dan AUC_RF per kelompok kelipatan 4 serta 13,14, dan 15 besar Dataset Privat

Nama Fitur	AUC_SF	AUC_RF
4 besar (x_6,x_5,x_3,x_4)	0.793393	0.919626
8 besar (x_6,x_5,x_3,x_4,y_5,y_4,x_7,y_7)	0.799972	0.93028
12 besar (x_6,x_5,x_3,x_4,y_5,y_4,x_7,y_7,y_6,y_8,x_2,y_2)	0.80317	0.930651
13 besar.	0.80298	0.93019
14 besar	0.80274	0.9295
15 besar	0.80068	0.92374
Semua Fitur Baru	0.79979224	0.90840720

Juga dapat dilihat dalam dataset privat hasil terbaik juga didapatkan 12 besar fitur peringkat pertama dengan nilai AUC-SF 0.80317 dan AUC-RF 0.930651. Berikut

adalah confusion matrix penggunaan 12 fitur baru peringkat pertama pada dataset Privat

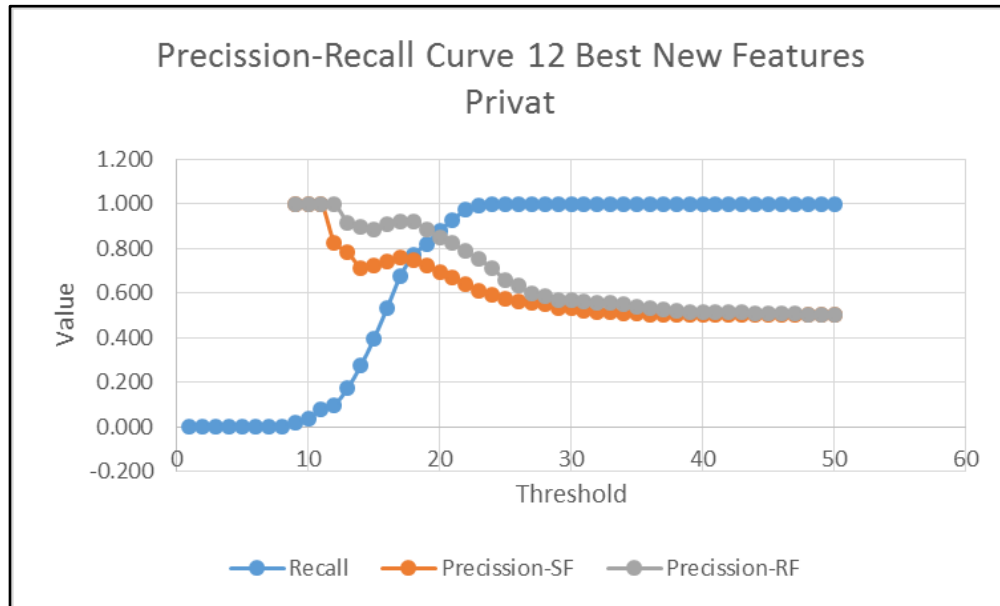


Gambar 4.15. AUC Dengan 12 Fitur Baru peringkat Tertinggi Dataset Privat

Tabel 4.25 Confusion Matrix Dengan 12 Fitur Peringkat Tetinggi Dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	190	190	0	190	0
2	0	190	190	0	190	0
3	0	190	190	0	190	0
4	0	190	190	0	190	0
5	0	190	190	0	190	0
6	0	190	190	0	190	0
7	0	190	190	0	190	0
8	0	190	190	0	190	0
9	4	186	190	0	190	0
10	7	183	190	0	190	0
11	15	175	190	0	190	0
12	19	171	186	4	190	0
13	33	157	181	9	187	3
14	52	138	169	21	184	6
15	75	115	161	29	180	10
16	101	89	155	35	180	10
17	128	62	149	41	179	11
18	146	44	140	50	177	13
19	156	34	131	59	170	20
20	167	23	116	74	160	30

21	176	14	102	88	152	38
22	185	5	85	105	140	50
23	188	2	71	119	128	62
24	189	1	61	129	113	77
25	190	0	48	142	92	98



Gambar 4.16. Kurva Precision-Recall Dengan 12 Fitur Baru peringkat Tertinggi Dataset Privat

Titik threshold optimal untuk kasus Skilled Forgery adalah 18.46767 dan titik threshold optimal untuk kasus Random Forgery adalah 20.35774. Berikut adalah hasil confusion matrix pada titik threshold tersebut.

Tabel 4.26 Confusion Matrix 12 Fitur Baru Threshold Optimal dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
18.46767	147	43	136	54	174	16
20.35774	170	20	113	77	157	33

4.3.4. Pemingkatan Fitur

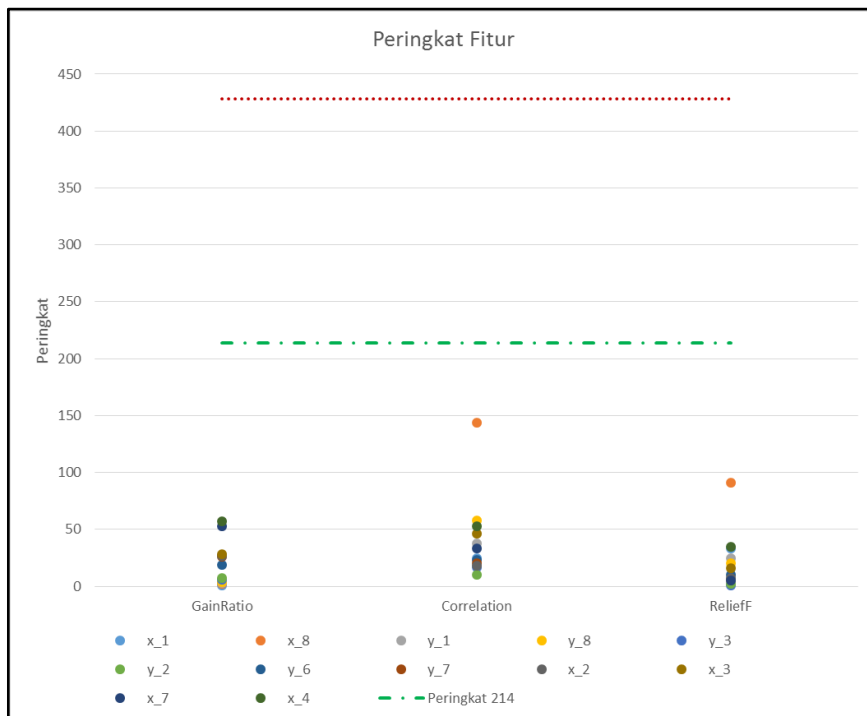
Untuk mengetahui peringkat fitur berdasarkan tingkat kepentingannya digunakan metode *Gain Ratio*, *Correlation*, dan *ReliefF Attribute Evaluator* yang ada di aplikasi Weka. Total ada 432 fitur yang digunakan dan dari 432 fitur tersebut, 12 fitur merupakan fitur baru yang sudah dipilih berdasarkan nilai AUC-SF dan memberikan kinerja yang paling baik. Dalam dataset SVC2004 12 fitur tersebut

antara lain (x_4,x_3,x_1,x_2,y_6,x_8, x_7,y_8,y_7,y_1,y_3,y_2). Berikut adalah hasil pemeringkatan 12 fitur baru tersebut dalam dataset SVC2004

Tabel 4.27 Peringkat 12 fitur baru dengan metode yang berbeda Dataset SVC2004

GainRatio		Correlation		ReliefF	
Nama Fitur	Peringkat	Nama Fitur	Peringkat	Nama Fitur	Peringkat
x_1	1	x_1	25	x_1	33
x_8	2	x_8	144	x_8	91
y_1	3	y_1	38	y_1	25
y_8	4	y_8	58	y_8	20
y_3	6	y_3	17	y_3	1
y_2	7	y_2	10	y_2	2
y_6	19	y_6	23	y_6	10
y_7	26	y_7	20	y_7	6
x_2	27	x_2	18	x_2	7
x_3	28	x_3	46	x_3	16
x_7	53	x_7	33	x_7	5
x_4	57	x_4	53	x_4	35

Untuk memudahkan memahami distribusi peringkat fitur, berikut adalah grafik yang merupakan konversi dari tabel di atas. Garis merah putus-putus adalah batas atas peringkat yaitu 432.



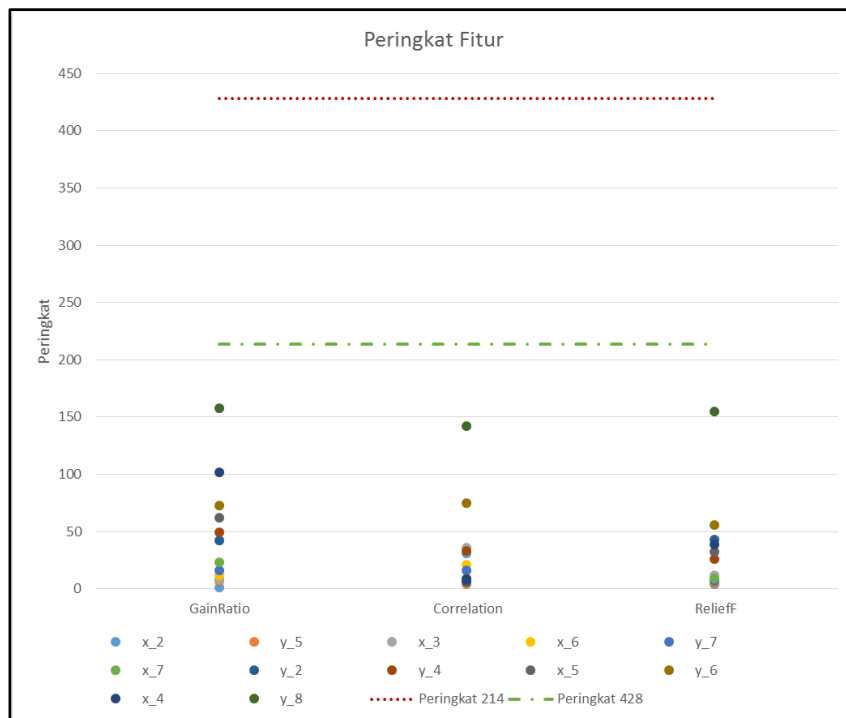
Gambar 4.17. Peringkat 12 fitur baru Dataset SVC2004 dengan tiga metode yang berbeda dalam bentuk grafik

Dalam dataset Privat 12 fitur tersebut antara lain x_6, x_5, x_3, x_4, y_5, y_4, x_7, y_7, y_6, y_8, x_2, y_2. Berikut adalah hasil peringkat 12 fitur baru tersebut dengan 3 metode yang berbeda

Tabel 4.28 Peringkat 12 fitur baru dengan metode yang berbeda Dataset Privat

GainRatio		Correlation		Relieff	
Nama Fitur	Peringkat	Nama Fitur	Peringkat	Nama Fitur	Peringkat
x_2	1	x_2	31	x_2	4
y_5	7	y_5	4	y_5	5
x_3	8	x_3	36	x_3	12
x_6	12	x_6	21	x_6	8
y_7	16	y_7	16	y_7	7
x_7	23	x_7	7	x_7	9
y_2	42	y_2	6	y_2	43
y_4	49	y_4	33	y_4	26
x_5	62	x_5	9	x_5	32
y_6	73	y_6	75	y_6	56
x_4	102	x_4	8	x_4	39
y_8	158	y_8	142	y_8	155

Berikut adalah peringkat fitur dalam bentuk grafik



Gambar 4.18. Peringkat 12 fitur baru Dataset Privat dengan tiga metode yang berbeda dalam bentuk grafik

4.4. Analisis Hasil Uji Coba

Recall, Precision, F-Measure, dan Accuracy merupakan nilai evaluasi pada satu titik threshold. Pada penelitian ini, nilai-nilai tersebut diambil di titik threshold optimal yang sudah dicari dengan pendekatan dimana nilai Recall sama dengan nilai Precision. Perlu menjadi catatan bahwa nilai Recall dan Precision diambil dari fungsi kurva hasil pendekatan, oleh karena itu nilai Recall tersebut tidak sama dengan nilai Recall aktual. Karena dalam penelitian ini hanya diambil 50 threshold, ada kemungkinan kesalahan kesimpulan atau bias apabila menggunakan nilai-nilai di atas pada threshold tertentu, karena mungkin ada titik threshold yang sebenarnya lebih baik namun terlewat dari 50 threshold, oleh karena itu nilai-nilai tersebut hanya digunakan sebagai informasi pelengkap sedangkan untuk menentukan sistem verifikasi lebih baik atau lebih buruk, digunakan AUC (Area Under Curve), dimana AUC merupakan nilai evaluasi yang lebih mendekati kebenaran karena menggunakan keseluruhan threshold. Nilai AUC memiliki interpretasi yaitu nilai kemungkinan tanda tangan asli yang dipilih secara acak terverifikasi sebagai asli daripada tanda tangan palsu yang dipilih secara acak. Artinya apabila nilai AUC lebih besar maka lebih baik karena kemungkinan True Positive nya lebih besar. Nilai AUC berguna khususnya dalam studi perbandingan antara dua sistem diagnostik. Apabila dua sistem dibandingkan, lebih disarankan menggunakan nilai AUC.

4.4.1. Perbandingan Dynamic Time Warping dan Manhattan Distance

Tabel 4.29 Perbandingan Kinerja DTW dan Manhattan pada dataset SVC2004

Pemalsuan Terlatih			
Jenis	DTW	Manhattan Distance	Selisih
Threshold Optimal	13.2990881	13.9553	-
TP FN	250 150	171 229	-
TN FP	631 169	716 84	-
Accuracy	84.1%	79.6%	4.5%
Recall	0.625	0.428	0.197
Precision	0.597	0.671	0.074
F-Measure	0.611	0.522	0.089
AUC	0.79568	0.82191	0.02623
Pemalsuan Acak			
Jenis	DTW	Manhattan Distance	Selisih
Threshold Optimal	15.5709825	18.5558	-
TP FN	345 55	364 36	-
TN FP	702 98	713 87	-
Accuracy	87.3%	89.8%	2.5%
Recall	0.863	0.910	0.047
Precision	0.779	0.807	0.028
F-Measure	0.819	0.855	0.036
AUC	0.93786	0.93815	0.00029

Pada threshold optimal dataset SVC2004 pemalsuan terlatih memiliki keunggulan nilai kinerja yang berbeda-beda. DTW memiliki akurasi lebih besar dengan selisih 4.5% dari Manhattan, Recall DTW lebih besar dengan selisih 0.197, Nilai F-Measure DTW juga lebih besar dengan selisih 0.089. Namun disini precision DTW lebih kecil dengan selisih 0.089. Sedangkan nilai kinerja yang lebih luas karena mencakup keseluruhan threshold yaitu AUC, metode DTW juga lebih kecil dengan selisih 0.02623. Berbeda dengan kasus pemalsuan terlatih, kasus pemalsuan acak menunjukkan keseragaman nilai di titik threshold optimal. Nilai akurasi DTW lebih kecil daripada Manhattan dengan selisih 2.5%, Recall lebih kecil dengan selisih 0.197, nilai Precision DTW lebih kecil dengan selisih 0.028, Nilai F-Measure juga lebih kecil dengan selisih 0.089. Nilai AUC DTW juga lebih kecil dengan selisih 0.02623.

Tabel 4.30 Perbandingan Kinerja DTW dan Manhattan pada dataset Privat

Pemalsuan Terlatih			
Jenis	DTW	Manhattan Distance	Selisih
Threshold Optimal	15.0417778	18.35086711	-
TP FN	123 67	145 45	-
TN FP	126 64	129 61	-
Accuracy	65.5%	72.1%	6.6%
Recall	0.647	0.763	0.116
Precision	0.658	0.704	0.046
F-Measure	0.653	0.732	0.079
AUC	0.72194	0.775540166	0.053600166
Pemalsuan Acak			
Jenis	DTW	Manhattan Distance	Selisih
Threshold Optimal	16.6650682	20.21781061	-
TP FN	161 29	164 26	-
TN FP	141 49	151 39	-
Accuracy	79.5%	82.9%	3.40%
Recall	0.847	0.863	0.016
Precision	0.767	0.808	0.041
F-Measure	0.805	0.835	0.030
AUC	0.8527424	0.900193906	0.047

Pada threshold optimal dataset Privat juga memiliki keseragaman pada nilai-nilai kinerjanya. Pada kasus pemalsuan terlatih akurasi DTW lebih kecil sebesar 6.6%. Nilai Recall DTW juga lebih kecil sebesar 0.116. Precision DTW juga lebih kecil sebesar 0.046. Nilai F-Measure lebih kecil dengan selisih 0.079. Dan kemudian nilai AUC DTW juga lebih kecil dengan selisih 0.0536. Pada kasus pemalsuan acak nilai-nilai kinerja DTW juga memiliki tren yang sama. Nilai akurasi DTW lebih kecil sebesar 3.40%, nilai Recall DTW lebih kecil dengan selisih 0.016. Nilai Precision DTW lebih kecil dengan selisih 0.041. Nilai F-Measure DTW lebih kecil sebesar 0.030. Dan yang terakhir nilai AUC DTW juga lebih kecil dengan selisih 0.047. Nilai yang menjadi prioritas adalah AUC, dapat dilihat bahwa nilai AUC DTW selalu lebih kecil. Dari hasil tersebut dapat disimpulkan bahwa DTW relatif lebih buruk apabila dibandingkan dengan Manhattan.

Hal yang menjadi alasan mengapa hal tersebut bisa terjadi dapat dijelaskan dalam penjelasan berikut. Dynamic Time Warping adalah algoritma untuk mengukur kesamaan ataupun ketidaksamaan antara dua deret temporal yang

bervariasi dalam waktu dan kecepatannya. Sebagai contoh, kesamaan pola berjalan dapat diukur menggunakan algoritma DTW, walaupun satu orang berjalan lebih cepat dari yang lain atau terjadi percepatan maupun perlambatan selama observasi. Artinya panjang vektor satu dengan yang lain bisa berbeda. Secara teori, memang DTW lebih tepat digunakan untuk menghitung jarak ketidaksamaan antara dua *time series* atau deret yang ada dimensi waktu, sedangkan Manhattan Distance lebih tepat digunakan untuk menghitung jarak ketidaksamaan antara dua deret tanpa ada dimensi waktu dengan panjang vektor yang sama. Dalam kasus ini, vektor fitur histogram merupakan deret vektor yang sudah tidak memiliki dimensi waktu, melainkan merupakan deret vektor dari distribusi bin dari histogram yang dibentuk.

4.4.2. Perbandingan Kinerja Fitur Baru

Tabel 4.31 Kinerja pada dataset SVC2004

Pemalsuan Terlatih			
Jenis	Dengan Fitur Baru	Tanpa Fitur Baru	Selisih
Threshold Optimal	15.90612845	13.9553	-
TP FN	259 141	171 229	-
TN FP	641 159	716 84	-
Accuracy	75%	79.6%	4.60%
Recall	0.648	0.428	0.22
Precision	0.620	0.671	0.051
F-Measure	0.633	0.522	0.111
AUC	0.828453	0.82191	0.00654
Pemalsuan Acak			
Jenis	Dengan Fitur Baru	Tanpa Fitur Baru	Selisih
Threshold Optimal	18.66426896	20.21781061	-
TP FN	356 44	164 26	-
TN FP	702 98	151 39	-
Accuracy	88.2%	82.9%	5.30%
Recall	0.890	0.863	0.027
Precision	0.784	0.808	0.024
F-Measure	0.834	0.835	0.001
AUC	0.955095313	0.900193906	0.0549

Pada threshold optimal dataset SVC2004 pemalsuan terlatih memiliki keunggulan nilai kinerja yang berbeda-beda. Sistem dengan penggunaan fitur baru memiliki akurasi lebih kecil dengan selisih 4.6%. Nilai recall dengan fitur baru lebih besar dengan selisih 0.22, Nilai precision dengan fitur baru lebih kecil

dengan selisih 0.051. Nilai F-Measure dengan fitur baru lebih besar dengan selisih 0.111. Nilai AUC dengan fitur baru lebih besar dengan selisih 0.001. Dalam kasus pemalsuan acak nilai-nilai kinerja lebih seragam. Akurasi dengan fitur baru lebih besar dengan selisih 5.3%. Nilai Recall lebih besar dengan selisih 0.027. Hanya nilai Precision dan F-Measure yang lebih kecil dengan selisih 0.024 dan 0.001, dan yang terakhir nilai yang paling menentukan nilai AUC lebih besar dengan selisih 0.0549.

Tabel 4.32 Kinerja pada dataset Privat

Pemalsuan Terlatih			
Jenis	Dengan Fitur Baru	Tanpa Fitur Baru	Selesai
Threshold Optimal	18.46767074	18.35086711	-
TP FN	147 43	145 45	-
TN FP	136 54	129 61	-
Accuracy	0.745	72.1%	2.40%
Recall	0.774	0.763	0.011
Precision	0.731	0.704	0.027
F-Measure	0.752	0.732	0.02
AUC	0.803171745	0.775540166	0.02763
Pemalsuan Acak			
Jenis	Dengan Fitur Baru	Tanpa Fitur Baru	Selesai
Threshold Optimal	20.35773703	20.21781061	-
TP FN	170 20	164 26	-
TN FP	157 33	151 39	-
Accuracy	86.1%	82.9%	3.20%
Recall	0.895	0.863	0.032
Precision	0.837	0.808	0.029
F-Measure	0.865	0.835	0.03
AUC	0.930651	0.900193906	0.03046

Pada threshold optimal dataset Privat juga memiliki keseragaman pada nilai-nilai kerjanya. Sistem dengan menggunakan fitur baru memiliki akurasi lebih besar dengan selisih 2.40%. Nilai recall lebih besar dengan selisih 0.011. Nilai precision lebih besar dengan nilai 0.027. Nilai F-Measure lebih besar dengan selisih 0.02. Kemudian nilai AUC juga lebih besar dengan selisih 0.02763. Pada kasus pemalsuan acak nilai-nilai kinerja juga memiliki tren yang sama. Akurasi sistem yang menggunakan fitur baru lebih besar dengan selisih 3.20%. Recall lebih besar dengan selisih 0.032. Precision lebih besar dengan selisih 0.010. F-Measure lebih besar dengan selisih 0.03. Dan nilai AUC juga lebih besar dengan nilai

0.03046. Nilai yang menjadi prioritas adalah AUC, dapat dilihat bahwa nilai AUC dengan fitur baru selalau lebih besar daripada sistem tanpa menggunakan fitur baru.

Dari hasil diatas diketahui bahwa kinerja sistem yang menggunakan fitur-fitur baru lebih baik daripada kinerja sistem tanpa menggunakan fitur-fitur baru. Dalam penelitian ini, fitur-fitur baru yang digunakan adalah histogram dari X dan Y dimana histogram tersebut merupakan gambaran distribusi nilai X dan Y dari tanda tangan. Nilai X dan Y tersebut berkaitan dengan bentuk tanda tangan. Aspek bentuk ini yang tidak digunakan dalam fitur-fitur histogram yang dilakukan pada penelitian sebelumnya, namun yang digunakan adalah turunan pertama dan turunan kedua dari X dan Y serta histogram dua dimensi hasil gabungan dari histogram-histogram di atas. Histogram turunan pertama dan turunan kedua merupakan gambaran distribusi kecepatan dan percepatan dari tanda tangan. Oleh karena itu, fitur distribusi X dan Y inilah yang memberikan pembeda sehingga verifikasi mampu membedakan tanda tangan asli dan palsu dengan lebih baik. Dengan adanya histogram X dan Y verifikasi menjadi lengkap dengan memperhatikan faktor bentuk, kecepatan, dan percepatan.

4.4.3. Pemeringkatan Fitur

Dari hasil pemeringkatan fitur dengan tiga metode pemeringkatan fitur dapat diketahui peringkat dari 12 fitur-fitur baru dalam 428 set fitur. Dalam dataset SVC2004, fitur-fitur tersebut antara lain x_4 , x_3 , x_1 , x_2 , y_6 , x_8 , x_7 , y_8 , y_7 , y_1 , y_3 , y_2 . Dengan metode Gain Ratio hasil peringkat yang didapatkan adalah 57, 28, 1, 27, 19, 2, 53, 4, 26, 3, 6, 7. Dengan metode Correlation, peringkat yang didapatkan adalah 53, 46, 25, 27, 23, 144, 33, 58, 20, 38, 17, 10. Dengan metode ReliefF, peringkat yang didapatkan adalah 35, 16, 33, 7, 91, 5, 20, 6, 25, 1, 2.

Dalam dataset Privat 12 fitur baru yang digunakan adalah x_6 , x_5 , x_3 , x_4 , y_5 , y_4 , x_7 , y_7 , y_6 , y_8 , x_2 , y_2 . Dengan metode Gain Ratio peringkat yang didapatkan adalah 12, 62, 8, 102, 7, 49, 23, 16, 73, 158, 1, 42. Dengan metode Correlation, peringkat yang didapatkan adalah 21, 9, 36, 8, 4, 33, 7, 75, 142, 31, 6. Dengan metode ReliefF, peringkat yang didapatkan adalah 8, 32, 12, 39, 5, 26, 9, 7, 56, 155, 4, 43. Secara keseluruhan, lebih dari atau sama dengan 50 persen dari

fitur-fitur baru berada pada peringkat 50 besar. Yang perlu diperhatikan juga bahwa 12 fitur tersebut tidak ada yang berada di peringkat yang lebih dari setengah dari total fitur dari semua metode pemeringkatan.

Walaupun secara peringkat tidak ada yang sama dengan ketiga metode tersebut, namun dapat dikatakan bahwa 12 fitur-fitur tersebut konsisten peringkatnya dengan selisih yang relatif kecil apabila dibandingkan dengan jumlah total keseluruhan fitur. Hal tersebut juga dapat dilihat secara visual melalui grafik scatter plot satu dimensi dari peringkat fitur. Fitur-fitur baru tersebut cenderung terkumpul pada satu cluster di peringkat awal. Hasil peringkat fitur ini juga menjelaskan mengapa sistem yang menggunakan 12 fitur baru memiliki kinerja yang lebih baik .

BAB 5

KESIMPULAN DAN SARAN

Bab ini menjelaskan kesimpulan dari hasil dan pembahasan yang dilakukan sesuai dengan skenario uji coba. Selain itu, dalam bab ini juga dijelaskan saran kemungkinan pengembangan lebih lanjut dari penelitian yang telah dilakukan.

5.1. Kesimpulan

Berdasarkan hasil uji coba dan pembahasan, dapat disimpulkan sebagai berikut:

1. Proses verifikasi tanda tangan dilakukan dengan beberapa tahap antara lain
 - a. Penyiapan data untuk ujicoba yaitu dengan menggunakan dataset SVC2004 tersedia di Internet dan dataset Privat yang dibuat dengan perangkat mobile.
 - b. Praproses dengan melakukan normalisasi waktu untuk dataset Privat dan Penggabungan garis untuk kedua dataset
 - c. Derivasi data (X, Y, Φ, R) menjadi (X^1, Y^1, Φ^1, R^1) dan (X^2, Y^2, Φ^2, R^2)
 - d. Ekstraksi Fitur histogram. Disiapkan dua fitur set, pertama fitur set dengan fitur baru, yang kedua fiturset tanpa fitur baru
 - e. Pembuatan template tanda tangan dari 10 data latih.
 - f. Verikasi dengan verifikator yang sudah dibangun. Ada dua verifikator yang dibangun. Yang pertama dengan menggunakan DTW dan yang kedua dengan menggunakan Manhattan Distance
2. Penggunaan Dynamic Time Warping sebagian besar menghasilkan kinerja yang lebih buruk daripada Manhattan Distance. Nilai AUC sistem verifikasi dengan metode DTW selalu lebih kecil daripada sistem verifikasi dengan metode Manhattan Distance. Pada dataset SVC2004 kasus pemalsuan terlatih nilai AUC DTW lebih buruk dengan selisih 0.02623. Pada kasus pemalsuan acak nilai AUC DTW juga lebih buruk dengan selisih 0.00029. Hal yang sama juga didapatkan pada dataset Privat. Pada kasus pemalsuan terlatih nilai AUC

DTW lebih kecil dengan selisih 0.053600166. Pada kasus pemalsuan acak nilai AUC DTW lebih kecil dengan selisih 0.047.

3. Penggunaan fitur-fitur baru yang diturunkan dari histogram X dan Y memberikan kinerja yang lebih baik. Nilai AUC sistem dengan fitur-fitur baru selalu lebih baik atau lebih besar. Pada dataset SVC2004 kasus pemalsuan terlatih nilai AUC sistem dengan fitur baru lebih besar dengan selisih 0.00654. Sedangkan pada kasus pemalsuan acak nilai AUC sistem dengan fitur-fitur baru lebih besar dengan selisih 0.0549. Pada dataset Privat kasus pemalsuan terlatih nilai AUC sistem dengan fitur baru lebih besar dengan selisih 0.024. Juga pada kasus pemalsuan acak nilai AUC lebih besar dengan selisih 0.08.
4. Fitur-fitur baru yang paling berpengaruh terhadap kinerja pada dataset SVC2004 adalah 12 fitur dengan urutan sebagai berikut $x_4, x_3, x_1, x_2, y_6, x_8, x_7, y_8, y_7, y_1, y_3, y_2$. Sedangkan fitur-fitur baru yang paling berpengaruh terhadap kinerja pada dataset Privat adalah 12 fitur dengan urutan sebagai berikut $x_6, x_5, x_3, x_4, y_5, y_4, x_7, y_7, y_6, y_8, x_2$, dan y_2 .
5. Fitur-fitur baru yang berpengaruh negatif karena mengurangi tingkat kinerja sistem dalam dataset SVC2004 adalah y_4, x_5, x_6 , dan y_5 . Fitur-fitur baru yang berpengaruh negatif dalam dataset Privat adalah x_1, y_1, y_3 , dan x_8 .
6. Dua belas fitur-fitur baru memiliki peringkat yang tinggi dan konsisten dengan menggunakan metode pemeringkatan yang berbeda. Tinggi karena 12 fitur berada di bawah peringkat 214 atau setengah dari 428 dan konsisten karena selisih antar peringkat kecil.
 - a Dalam dataset SVC2004, fitur-fitur tersebut antara lain $x_4, x_3, x_1, x_2, y_6, x_8, x_7, y_8, y_7, y_1, y_3, y_2$. Dengan metode Gain Ratio hasil peringkat yang didapatkan adalah 57, 28, 1, 27, 19, 2, 53, 4, 26, 3, 6, 7. Dengan metode Correlation, peringkat yang didapatkan adalah 53, 46, 25, 27, 23, 144, 33, 58, 20, 38, 17, 10. Dengan metode ReliefF, peringkat yang didapatkan adalah 35, 16, 33, 7, 91, 5, 20, 6, 25, 1.
 - b Dalam dataset Privat 12 fitur baru yang digunakan adalah $x_6, x_5, x_3, x_4, y_5, y_4, x_7, y_7, y_6, y_8, x_2, y_2$. Dengan metode Gain Ratio peringkat yang didapatkan adalah 12, 62, 8, 102, 7, 49, 23, 16, 73, 158, 1, 42. Dengan metode Correlation, peringkat yang didapatkan adalah 21, 9,

36, 8, 4, 33, 7, 75, 142, 31, 6. Dengan metode ReliefF, peringkat yang didapatkan adalah 8, 32, 12, 39, 5, 26, 9, 7, 56, 155, 4, 43

5.2. Saran

Saran-saran yang dapat diberikan berkaitan dengan hasil uji coba dan pembahasan pada penelitian ini adalah:

1. Tanda tangan privat pada penelitian ini diambil hanya dengan menggunakan satu perangkat mobile. Kedepan bisa dicoba pengambilan tanda tangan dengan perangkat yang lebih bervariasi dan multiplatform (Android dan IOS).
2. Penelitian ini hanya terbatas pada penggunaan fitur-fitur yang diturunkan dari histogram. Penelitian selanjutnya bisa dikombinasikan dengan fitur-fitur baru. Sebagai contoh fitur yang bisa diambil dari pemrosesan sinyal digital seperti energi dari sinyal yang mungkin dapat membedakan tanda tangan asli atau palsu dengan lebih baik.

(lembar ini sengaja dikosongkan)

DAFTAR PUSTAKA

- Aha, D. W., Kibler, D., & Albert, M. K. (1991). Instance-based learning algorithms. *Machine Learning*, 6(1), 37–66. <https://doi.org/10.1007/BF00153759>
- Al-Omari, Y. M., Abdullah, S. N. H. S., & Omar, K. (2011). State-of-the-art in offline signature verification system. *2011 International Conference on Pattern Analysis and Intelligence Robotics*, 1(June), 59–64. <https://doi.org/10.1109/ICPAIR.2011.5976912>
- Brault, J.-J., & Plamondon, R. (1989). How to detect problematic signers for automatic signature verification. In *Proceedings. International Carnahan Conference on Security Technology* (Vol. 1, pp. 127–132). IEEE. <https://doi.org/10.1109/CCST.1989.751966>
- Brault, J. J., & Plamondon, R. (1993). A Complexity Measure of Handwritten Curves: Modeling of Dynamic Signature Forgery. *IEEE Transactions on Systems, Man and Cybernetics*, 23(2), 400–413. <https://doi.org/10.1109/21.229453>
- Callan, J. P., Fawcett, T. E., & Rissland, E. L. (1991). CABOT: An Adaptive Approach to Case-based Search. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence - Volume 2* (pp. 803–808). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. Retrieved from <http://dl.acm.org/citation.cfm?id=1631552.1631581>
- Chang, W., & Shin, J. (2007). Modified Dynamic Time Warping for Stroke-Based On-line Signature Verification. *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2*, 724–728. <https://doi.org/10.1109/ICDAR.2007.4377010>
- Corradin, R. S. (2008). *Signature verification in consignment notes*. VU University: Amsterdam.
- Crane, H. D., & Ostrem, J. S. (1983). Automatic signature verification using a three-axis force-sensitive pen. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(3), 329–337. <https://doi.org/10.1109/TSMC.1983.6313165>

- Dimauro, G., Impedovo, S., & Pirlo, G. (1994). Component-Oriented Algorithms for Signature Verification. *International Journal of Pattern Recognition and Artificial Intelligence*, 8(3), 771. <https://doi.org/citeulike-article-id:950695>
- Dolfing, J. G. A. (1998). Handwriting recognition and verification, a Hidden Markov approach. *Philips Electronics N.V.* <https://doi.org/10.6100/IR517815>
- Dolfing, J. G. A., Aarts, E. H. L., & van Oosterhout, J. J. G. M. (1998). On-line signature verification with hidden Markov models. *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No.98EX170)*, 2, 1309–1312. <https://doi.org/10.1109/ICPR.1998.711942>
- Fallah, A., Jamaati, M., & Soleamani, A. (2011). A new online signature verification system based on combining Mellin transform, MFCC and neural network. *Digital Signal Processing*, 21(2), 404–416. <https://doi.org/https://doi.org/10.1016/j.dsp.2010.09.004>
- Faundez-Zanuy, M. (2007). On-line signature recognition based on VQ-DTW. *Pattern Recognition*, 40(3), 981–992. <https://doi.org/https://doi.org/10.1016/j.patcog.2006.06.007>
- Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), 861–874. <https://doi.org/https://doi.org/10.1016/j.patrec.2005.10.010>
- Feng, H., & Wah, C. C. (2003). Online signature verification using a new extreme points warping technique. *Pattern Recognition Letters*, 24(16), 2943–2951. [https://doi.org/10.1016/S0167-8655\(03\)00155-7](https://doi.org/10.1016/S0167-8655(03)00155-7)
- Fierrez-Aguilar, J., Nanni, L., Lopez-Peñalba, J., Ortega-Garcia, J., & Maltoni, D. (2005). An On-Line Signature Verification System Based on Fusion of Local and Global Information. In *Audio- and Video-Based Biometric Person Authentication* (Vol. 3546, pp. 523–532). https://doi.org/10.1007/11527923_54
- Fierrez-Aguilar, J., Ortega-Garcia, J., & Gonzalez-Rodriguez, J. (2005). Target dependent score normalization techniques and their application to signature verification. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35(3), 1556–1558. <https://doi.org/10.1109/TSMCC.2005.848181>
- Fierrez, J., Ortega-Garcia, J., Ramos, D., & Gonzalez-Rodriguez, J. (2007). HMM-

- based on-line signature verification: Feature extraction and signature modeling. *Pattern Recognition Letters*, 28(16), 2325–2334. <https://doi.org/10.1016/j.patrec.2007.07.012>
- Fischer, J. E., Bachmann, L. M., & Jaeschke, R. (2003). A readers' guide to the interpretation of diagnostic test properties: clinical example of sepsis. *Intensive Care Medicine*, 29(7), 1043–1051. <https://doi.org/10.1007/s00134-003-1761-8>
- Garcia-Salicetti, S. (2007). Biosecure reference systems for on-line signature verification: A study of complementarity. *Annales Des ...*, (January 2007), 36–61. <https://doi.org/10.1007/BF03253249>
- Garcia-Salicetti, S., Houmani, N., Ly-Van, B., Dorizzi, B., Alonso-Fernandez, F., Fierrez, J., ... Scheidat, T. (2009). Online Handwritten Signature Verification. In D. Petrovska-Delacrétaz, B. Dorizzi, & G. Chollet (Eds.), *Guide to Biometric Reference Systems and Performance Evaluation* (pp. 125–165). London: Springer London. https://doi.org/10.1007/978-1-84800-292-0_6
- Garcia-Salicetti, S., Houmani, N., Ly-Van, B., Dorizzi, B., Alonso-Fernandez, F., Fierrez, J., ... Scheidat, T. (2009). *Online Handwritten Signature Verification Guide to biometric reference systems and performance evaluation. Guide to Biometric Reference Systems and Performance Evaluation*. <https://doi.org/10.1007/978-1-84800-292-0>
- GURRALA, K. K. (2011). *Online Signature Verification Techniques*. National Institute Of Technology: Rourkela.
- Hall, M. A., & Holmes, G. (2003). Benchmarking attribute selection techniques for discrete class data mining. *IEEE Transactions on Knowledge and Data Engineering*, 15(6), 1437–1447. <https://doi.org/10.1109/TKDE.2003.1245283>
- Hangai, S., Yamanaka, S., & Hammamoto, T. (2000). Writer verification using altitude and direction of pen movement. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000* (Vol. 3, pp. 479–482). IEEE Comput. Soc. <https://doi.org/10.1109/ICPR.2000.903588>
- Hao, F. (2011). *Cryptosystem with private key generation from dynamic properties of human hand signature*. Nanyang Technological University.
- Jain, A. K., Griess, F. D., & Connell, S. D. (2002). On-line signature verification.

Pattern Recognition, 35(12), 2963–2972. [https://doi.org/10.1016/S0031-3203\(01\)00240-0](https://doi.org/10.1016/S0031-3203(01)00240-0)

Karypis, G., Aggarwal, R., Kumar, V., & Shekhar, S. (1999). Multilevel hypergraph partitioning: applications in VLSI domain. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 7(1), 69–79. <https://doi.org/10.1109/92.748202>

Kashi, R., Hu, J., Nelson, W. L., & Turin, W. (1998). A Hidden Markov Model approach to online handwritten signature verification. *International Journal on Document Analysis and Recognition*, 1(2), 102–109. <https://doi.org/10.1007/s100320050010>

Ketabdar, H., Richiardi, J., & Drygajlo, A. (2005). Global feature selection for on-line signature verification.

Kholmatov, A., & Yanikoglu, B. (2005). Identity authentication using improved online signature verification method. *Pattern Recognition Letters*, 26(15), 2400–2408. <https://doi.org/10.1016/j.patrec.2005.04.017>

Kholmatov, A., & Yanikoglu, B. (2009). SUSIG: An on-line signature database, associated protocols and benchmark results. *Pattern Analysis and Applications*, 12(3), 227–236. <https://doi.org/10.1007/s10044-008-0118-x>

Kira, K., & Rendell, L. A. (1992). A Practical Approach to Feature Selection. In *Proceedings of the Ninth International Workshop on Machine Learning* (pp. 249–256). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. Retrieved from <http://dl.acm.org/citation.cfm?id=141975.142034>

Komiya, Y., & Matsumoto, T. (1999). On-line pen input signature verification ppi (pen-position/pen-pressure/pen-inclination). ... *and Cybernetics, 1999. IEEE SMC'99* ..., 41–46. Retrieved from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=812373

Lee, L. L., Berger, T., & Aviczer, E. (1996). Reliable on-line human signature verification systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(6), 643–647. <https://doi.org/10.1109/34.506415>

Nelson, W., Turin, W., & Hastie, T. (1994). Statistical methods for on-line signature verification. *Intl. Journal of Pattern Recognition and Artificial Intell*, 8, 749–770. <https://doi.org/10.1142/S0218001494000395>

- Ortega-Garcia, J., Fierrez-Aguilar, J., Simon, D., Gonzalez, J., Faundez-Zanuy, M., Espinosa, V., ... Moro, Q.-I. (2003). MCYT baseline corpus: a bimodal biometric database. *IEE Proceedings - Vision, Image, and Signal Processing*, *150*(6), 395. <https://doi.org/10.1049/ip-vis:20031078>
- Plackett, R. L. (1983). Karl pearson and the chi-squared test, *51*, 59–72.
- Plamondon, R. (1993). Looking at handwriting generation from a velocity control perspective. *Acta Psychologica*, *82*(1–3), 89–101. [https://doi.org/10.1016/0001-6918\(93\)90006-D](https://doi.org/10.1016/0001-6918(93)90006-D)
- Qiao, Y., Liu, J., & Tanf, X. (2007). Offline Signature Verification Using Online Handwriting Registration. *CVPR*. <https://doi.org/10.1109/CVPR.2007.383263>
- Quinlan, J. R. (1986). Induction of Decision Trees. *Mach. Learn.*, *1*(1), 81–106. <https://doi.org/10.1023/A:1022643204877>
- Rabiner, L., & Juang, B.-H. (1993). *Fundamentals of Speech Recognition*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.
- Richiardi, J., Ketabdar, H. K. H., & Drygajlo, a. (2005). Local and global feature selection for on-line signature verification. *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*, 0–4. <https://doi.org/10.1109/ICDAR.2005.152>
- Rigoll, G., & Kosmala, A. (1998). A systematic comparison between on-line and off-line methods for signature verification with hidden Markov models. *Proceedings Fourteenth International Conference on Pattern Recognition*, 1–3. <https://doi.org/10.1109/icpr.1998.712066>
- Sae-bae, N., & Memon, N. (2014). Online Signature Verification on Mobile Devices, *9*(6), 933–947. <https://doi.org/10.1109/TIFS.2014.2316472>
- Salvador, S., & Chan, P. (2007). Toward Accurate Dynamic Time Warping in Linear Time and Space. *Intell. Data Anal.*, *11*(5), 561–580. Retrieved from <http://dl.acm.org/citation.cfm?id=1367985.1367993>
- Schiele, B., & Crowley, J. L. (1996). Object Recognition Using Multidimensional Receptive Field Histograms. In *Proceedings of the 4th European Conference on Computer Vision-Volume I - Volume I* (pp. 610–619). London, UK, UK: Springer-Verlag. Retrieved from <http://dl.acm.org/citation.cfm?id=645309.648952>

- Swets, J. A. (1979). ROC analysis applied to the evaluation of medical imaging techniques. *Investigative Radiology*, 14(2), 109–121.
- Taherzadeh, G., Karimi, R., Ghobadi, A., & Beh, H. M. (2011). Evaluation of online signature verification features. *Advanced Communication Technology (ICACT), 2011 13th International Conference on*, (July), 772–777.
- Wirocius, M., Ramel, J. Y., & Vincent, N. (2004). Selection of points for on-line signature comparison. *Proceedings - International Workshop on Frontiers in Handwriting Recognition, IWFHR*, 503–508.
<https://doi.org/10.1109/IWFHR.2004.92>
- Yanikoglu, B., & Kholmatov, A. (2009). Online signature verification using fourier descriptors. *Eurasip Journal on Advances in Signal Processing*, 2009.
<https://doi.org/10.1155/2009/260516>
- Yeung, D., Chang, H., Xiong, Y., & George, S. (2004). SVC2004: First international signature verification competition. *Biometric ...*, 16–22.
https://doi.org/10.1007/978-3-540-25948-0_3

(lembar ini sengaja dikosongkan)

LAMPIRAN A

Lampiran ini berisi confusion matrix dan hasil performa verifikasi tanda tangan lengkap sejumlah 50 threshold.

A.1 Perbandingan Dynamic Time Warping dan Manhattan Distance

A.1.1 Kinerja DTW pada Dataset SVC2004

Berikut ini adalah tabel confusion matrix DTWarping pada Dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	400	800	0	800	0
2	0	400	800	0	800	0
3	0	400	800	0	800	0
4	0	400	800	0	800	0
5	0	400	800	0	800	0
6	1	399	800	0	800	0
7	7	393	800	0	800	0
8	15	385	800	0	800	0
9	33	367	792	8	800	0
10	57	343	778	22	798	2
11	96	304	749	51	792	8
12	169	231	701	99	779	21
13	231	169	648	152	769	31
14	286	114	595	205	741	59
15	325	75	528	272	713	87
16	356	44	448	352	684	116
17	374	26	361	439	658	142
18	384	16	294	506	610	190
19	390	10	236	564	566	234
20	395	5	185	615	511	289
21	395	5	137	663	446	354
22	398	2	97	703	402	398
23	400	0	75	725	362	438
24	400	0	54	746	309	491
25	400	0	42	758	267	533
26	400	0	30	770	232	568
27	400	0	22	778	188	612
28	400	0	18	782	157	643
29	400	0	13	787	132	668
30	400	0	8	792	105	695
31	400	0	5	795	90	710
32	400	0	4	796	78	722
33	400	0	4	796	64	736
34	400	0	2	798	46	754
35	400	0	0	800	32	768
36	400	0	0	800	28	772
37	400	0	0	800	24	776
38	400	0	0	800	22	778
39	400	0	0	800	15	785
40	400	0	0	800	14	786
41	400	0	0	800	13	787
42	400	0	0	800	11	789
43	400	0	0	800	7	793
44	400	0	0	800	7	793
45	400	0	0	800	6	794
46	400	0	0	800	4	796
47	400	0	0	800	4	796
48	400	0	0	800	3	797
49	400	0	0	800	2	798
50	400	0	0	800	0	800

Berikut ini adalah tabel kinerja DTW dataset SVC2004

Thr	Recal	FPR-SF	FPR-RF	Precision-SF	F-Measure-SF	Acc-SF	Preci-RF	F-Meas-RF	Acc-RF
1	0.000	0.000	0.000	-	-	0.667	-	-	0.667
2	0.000	0.000	0.000	-	-	0.667	-	-	0.667
3	0.000	0.000	0.000	-	-	0.667	-	-	0.667
4	0.000	0.000	0.000	-	-	0.667	-	-	0.667
5	0.000	0.000	0.000	-	-	0.667	-	-	0.667
6	0.003	0.000	0.000	1.000	0.005	0.668	1.000	0.005	0.668
7	0.018	0.000	0.000	1.000	0.034	0.673	1.000	0.034	0.673
8	0.038	0.000	0.000	1.000	0.072	0.679	1.000	0.072	0.679
9	0.083	0.010	0.000	0.805	0.150	0.688	1.000	0.152	0.694
10	0.143	0.028	0.003	0.722	0.238	0.696	0.966	0.248	0.713
11	0.240	0.064	0.010	0.653	0.351	0.704	0.923	0.381	0.740
12	0.423	0.124	0.026	0.631	0.506	0.725	0.889	0.573	0.790
13	0.578	0.190	0.039	0.603	0.590	0.733	0.882	0.698	0.833
14	0.715	0.256	0.074	0.582	0.642	0.734	0.829	0.768	0.856
15	0.813	0.340	0.109	0.544	0.652	0.711	0.789	0.800	0.865
16	0.890	0.440	0.145	0.503	0.643	0.670	0.754	0.817	0.867
17	0.935	0.549	0.178	0.460	0.617	0.613	0.725	0.817	0.860
18	0.960	0.633	0.238	0.431	0.595	0.565	0.669	0.789	0.828
19	0.975	0.705	0.293	0.409	0.576	0.522	0.625	0.762	0.797
20	0.988	0.769	0.361	0.391	0.560	0.483	0.577	0.729	0.755
21	0.988	0.829	0.443	0.373	0.542	0.443	0.527	0.688	0.701
22	0.995	0.879	0.498	0.361	0.530	0.413	0.500	0.666	0.667
23	1.000	0.906	0.548	0.356	0.525	0.396	0.477	0.646	0.635
24	1.000	0.933	0.614	0.349	0.517	0.378	0.449	0.620	0.591
25	1.000	0.948	0.666	0.345	0.513	0.368	0.429	0.600	0.556
26	1.000	0.963	0.710	0.342	0.510	0.358	0.413	0.585	0.527
27	1.000	0.973	0.765	0.340	0.507	0.352	0.395	0.567	0.490
28	1.000	0.978	0.804	0.338	0.506	0.348	0.384	0.554	0.464
29	1.000	0.984	0.835	0.337	0.504	0.344	0.375	0.545	0.443
30	1.000	0.990	0.869	0.336	0.503	0.340	0.365	0.535	0.421
31	1.000	0.994	0.888	0.335	0.502	0.338	0.360	0.530	0.408
32	1.000	0.995	0.903	0.334	0.501	0.337	0.357	0.526	0.398
33	1.000	0.995	0.920	0.334	0.501	0.337	0.352	0.521	0.387
34	1.000	0.998	0.943	0.334	0.501	0.335	0.347	0.515	0.372
35	1.000	1.000	0.960	0.333	0.500	0.333	0.342	0.510	0.360
36	1.000	1.000	0.965	0.333	0.500	0.333	0.341	0.509	0.357
37	1.000	1.000	0.970	0.333	0.500	0.333	0.340	0.508	0.353
38	1.000	1.000	0.973	0.333	0.500	0.333	0.340	0.507	0.352
39	1.000	1.000	0.981	0.333	0.500	0.333	0.338	0.505	0.346
40	1.000	1.000	0.983	0.333	0.500	0.333	0.337	0.504	0.345
41	1.000	1.000	0.984	0.333	0.500	0.333	0.337	0.504	0.344
42	1.000	1.000	0.986	0.333	0.500	0.333	0.336	0.503	0.343
43	1.000	1.000	0.991	0.333	0.500	0.333	0.335	0.502	0.339
44	1.000	1.000	0.991	0.333	0.500	0.333	0.335	0.502	0.339
45	1.000	1.000	0.993	0.333	0.500	0.333	0.335	0.502	0.338
46	1.000	1.000	0.995	0.333	0.500	0.333	0.334	0.501	0.337
47	1.000	1.000	0.995	0.333	0.500	0.333	0.334	0.501	0.337
48	1.000	1.000	0.996	0.333	0.500	0.333	0.334	0.501	0.336
49	1.000	1.000	0.998	0.333	0.500	0.333	0.334	0.501	0.335
50	1.000	1.000	1.000	0.333	0.500	0.333	0.333	0.500	0.333

A.1.2 Kinerja Manhattan Distance pada Dataset SVC2004

Berikut ini adalah tabel confusion matrix Manhattan Distance pada dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	400	800	0	800	0
2	0	400	800	0	800	0
3	0	400	800	0	800	0
4	0	400	800	0	800	0
5	0	400	800	0	800	0
6	0	400	800	0	800	0
7	0	400	800	0	800	0
8	6	394	800	0	800	0
9	16	384	799	1	800	0
10	25	375	792	8	800	0
11	58	342	783	17	799	1
12	76	324	766	34	796	4
13	120	280	742	58	795	5
14	173	227	715	85	783	17
15	224	176	675	125	775	25
16	273	127	625	175	762	38
17	314	86	566	234	748	52
18	348	52	506	294	730	70
19	370	30	442	358	702	98
20	385	15	386	414	676	124
21	396	4	326	474	650	150
22	397	3	273	527	622	178
23	398	2	229	571	578	222
24	399	1	186	614	539	261
25	399	1	153	647	500	300
26	399	1	133	667	460	340
27	399	1	103	697	428	372
28	400	0	87	713	391	409
29	400	0	64	736	347	453
30	400	0	46	754	313	487
31	400	0	32	768	280	520
32	400	0	23	777	249	551
33	400	0	15	785	216	584
34	400	0	11	789	192	608
35	400	0	9	791	169	631
36	400	0	7	793	153	647
37	400	0	5	795	128	672
38	400	0	4	796	106	694
39	400	0	2	798	98	702
40	400	0	2	798	86	714
41	400	0	0	800	75	725
42	400	0	0	800	66	734
43	400	0	0	800	57	743
44	400	0	0	800	51	749
45	400	0	0	800	39	761
46	400	0	0	800	31	769
47	400	0	0	800	27	773
48	400	0	0	800	22	778
49	400	0	0	800	17	783
50	400	0	0	800	16	784

Berikut ini adalah tabel kinerja Mahattan pada dataset SVC2004

Thr	Recal	FPR-SF	FPR-RF	Precision-SF	F-Measure-SF	Acc-SF	Preci-RF	F-Meas-RF	Acc-RF
1	0.000	0.000	0.000	-	-	0.667	-	-	0.667
2	0.000	0.000	0.000	-	-	0.667	-	-	0.667
3	0.000	0.000	0.000	-	-	0.667	-	-	0.667
4	0.000	0.000	0.000	-	-	0.667	-	-	0.667
5	0.000	0.000	0.000	-	-	0.667	-	-	0.667
6	0.000	0.000	0.000	-	-	0.667	-	-	0.667
7	0.000	0.000	0.000	-	-	0.667	-	-	0.667
8	0.015	0.000	0.000	1.000	0.030	0.672	1.000	0.030	0.672
9	0.040	0.001	0.000	0.941	0.077	0.679	1.000	0.077	0.680
10	0.063	0.010	0.000	0.758	0.115	0.681	1.000	0.118	0.688
11	0.145	0.021	0.001	0.773	0.244	0.701	0.983	0.253	0.714
12	0.190	0.043	0.005	0.691	0.298	0.702	0.950	0.317	0.727
13	0.300	0.073	0.006	0.674	0.415	0.718	0.960	0.457	0.763
14	0.433	0.106	0.021	0.671	0.526	0.740	0.911	0.586	0.797
15	0.560	0.156	0.031	0.642	0.598	0.749	0.900	0.690	0.833
16	0.683	0.219	0.048	0.609	0.644	0.748	0.878	0.768	0.863
17	0.785	0.293	0.065	0.573	0.662	0.733	0.858	0.820	0.885
18	0.870	0.368	0.088	0.542	0.668	0.712	0.833	0.851	0.898
19	0.925	0.448	0.123	0.508	0.656	0.677	0.791	0.853	0.893
20	0.963	0.518	0.155	0.482	0.642	0.643	0.756	0.847	0.884
21	0.990	0.593	0.188	0.455	0.624	0.602	0.725	0.837	0.872
22	0.993	0.659	0.223	0.430	0.600	0.558	0.690	0.814	0.849
23	0.995	0.714	0.278	0.411	0.581	0.523	0.642	0.780	0.813
24	0.998	0.768	0.326	0.394	0.565	0.488	0.605	0.753	0.782
25	0.998	0.809	0.375	0.381	0.552	0.460	0.571	0.726	0.749
26	0.998	0.834	0.425	0.374	0.544	0.443	0.540	0.701	0.716
27	0.998	0.871	0.465	0.364	0.533	0.418	0.518	0.681	0.689
28	1.000	0.891	0.511	0.359	0.529	0.406	0.494	0.662	0.659
29	1.000	0.920	0.566	0.352	0.521	0.387	0.469	0.638	0.623
30	1.000	0.943	0.609	0.347	0.515	0.372	0.451	0.622	0.594
31	1.000	0.960	0.650	0.342	0.510	0.360	0.435	0.606	0.567
32	1.000	0.971	0.689	0.340	0.507	0.353	0.421	0.592	0.541
33	1.000	0.981	0.730	0.338	0.505	0.346	0.407	0.578	0.513
34	1.000	0.986	0.760	0.336	0.503	0.343	0.397	0.568	0.493
35	1.000	0.989	0.789	0.336	0.503	0.341	0.388	0.559	0.474
36	1.000	0.991	0.809	0.335	0.502	0.339	0.382	0.553	0.461
37	1.000	0.994	0.840	0.335	0.502	0.338	0.373	0.543	0.440
38	1.000	0.995	0.868	0.334	0.501	0.337	0.366	0.535	0.422
39	1.000	0.998	0.878	0.334	0.501	0.335	0.363	0.533	0.415
40	1.000	0.998	0.893	0.334	0.501	0.335	0.359	0.528	0.405
41	1.000	1.000	0.906	0.333	0.500	0.333	0.356	0.525	0.396
42	1.000	1.000	0.918	0.333	0.500	0.333	0.353	0.522	0.388
43	1.000	1.000	0.929	0.333	0.500	0.333	0.350	0.518	0.381
44	1.000	1.000	0.936	0.333	0.500	0.333	0.348	0.516	0.376
45	1.000	1.000	0.951	0.333	0.500	0.333	0.345	0.512	0.366
46	1.000	1.000	0.961	0.333	0.500	0.333	0.342	0.510	0.359
47	1.000	1.000	0.966	0.333	0.500	0.333	0.341	0.509	0.356
48	1.000	1.000	0.973	0.333	0.500	0.333	0.340	0.507	0.352
49	1.000	1.000	0.979	0.333	0.500	0.333	0.338	0.505	0.348
50	1.000	1.000	0.980	0.333	0.500	0.333	0.338	0.505	0.347

A.1.3 Kinerja DTW pada Dataset Privat

Berikut ini adalah tabel confusion matrix DTW pada Dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	190	190	0	190	0
2	0	190	190	0	190	0
3	0	190	190	0	190	0
4	0	190	190	0	190	0
5	0	190	190	0	190	0
6	0	190	190	0	190	0
7	1	189	190	0	190	0
8	11	179	190	0	190	0
9	20	170	186	4	190	0
10	30	160	180	10	189	1
11	46	144	169	21	185	5
12	59	131	155	35	183	7
13	73	117	146	44	177	13
14	89	101	137	53	169	21
15	123	67	126	64	157	33
16	149	41	114	76	147	43
17	163	27	99	91	134	56
18	174	16	87	103	119	71
19	179	11	74	116	105	85
20	186	4	66	124	92	98
21	187	3	53	137	81	109
22	189	1	38	152	76	114
23	190	0	28	162	67	123
24	190	0	21	169	57	133
25	190	0	16	174	51	139
26	190	0	15	175	45	145
27	190	0	11	179	40	150
28	190	0	6	184	35	155
29	190	0	3	187	31	159
30	190	0	1	189	28	162
31	190	0	0	190	23	167
32	190	0	0	190	22	168
33	190	0	0	190	22	168
34	190	0	0	190	18	172
35	190	0	0	190	15	175
36	190	0	0	190	11	179
37	190	0	0	190	11	179
38	190	0	0	190	8	182
39	190	0	0	190	6	184
40	190	0	0	190	3	187
41	190	0	0	190	2	188
42	190	0	0	190	2	188
43	190	0	0	190	1	189
44	190	0	0	190	1	189
45	190	0	0	190	0	190
46	190	0	0	190	0	190
47	190	0	0	190	0	190
48	190	0	0	190	0	190
49	190	0	0	190	0	190
50	190	0	0	190	0	190

Berikut adalah tabel kinerja DTW pada dataset Privat

Thr	Recal	FPR-SF	FPR-RF	Precision-SF	F-Measure-SF	Acc-SF	Preci-RF	F-Meas-RF	Acc-RF
1	0.000	0.000	0.000	-	-	0.500	-	-	0.500
2	0.000	0.000	0.000	-	-	0.500	-	-	0.500
3	0.000	0.000	0.000	-	-	0.500	-	-	0.500
4	0.000	0.000	0.000	-	-	0.500	-	-	0.500
5	0.000	0.000	0.000	-	-	0.500	-	-	0.500
6	0.000	0.000	0.000	-	-	0.500	-	-	0.500
7	0.005	0.000	0.000	1.000	0.010	0.503	1.000	0.010	0.503
8	0.058	0.000	0.000	1.000	0.109	0.529	1.000	0.109	0.529
9	0.105	0.021	0.000	0.833	0.187	0.542	1.000	0.190	0.553
10	0.158	0.053	0.005	0.750	0.261	0.553	0.968	0.271	0.576
11	0.242	0.111	0.026	0.687	0.358	0.566	0.902	0.382	0.608
12	0.311	0.184	0.037	0.628	0.415	0.563	0.894	0.461	0.637
13	0.384	0.232	0.068	0.624	0.476	0.576	0.849	0.529	0.658
14	0.468	0.279	0.111	0.627	0.536	0.595	0.809	0.593	0.679
15	0.647	0.337	0.174	0.658	0.653	0.655	0.788	0.711	0.737
16	0.784	0.400	0.226	0.662	0.718	0.692	0.776	0.780	0.779
17	0.858	0.479	0.295	0.642	0.734	0.689	0.744	0.797	0.782
18	0.916	0.542	0.374	0.628	0.745	0.687	0.710	0.800	0.771
19	0.942	0.611	0.447	0.607	0.738	0.666	0.678	0.789	0.747
20	0.979	0.653	0.516	0.600	0.744	0.663	0.655	0.785	0.732
21	0.984	0.721	0.574	0.577	0.728	0.632	0.632	0.770	0.705
22	0.995	0.800	0.600	0.554	0.712	0.597	0.624	0.767	0.697
23	1.000	0.853	0.647	0.540	0.701	0.574	0.607	0.755	0.676
24	1.000	0.889	0.700	0.529	0.692	0.555	0.588	0.741	0.650
25	1.000	0.916	0.732	0.522	0.686	0.542	0.578	0.732	0.634
26	1.000	0.921	0.763	0.521	0.685	0.539	0.567	0.724	0.618
27	1.000	0.942	0.789	0.515	0.680	0.529	0.559	0.717	0.605
28	1.000	0.968	0.816	0.508	0.674	0.516	0.551	0.710	0.592
29	1.000	0.984	0.837	0.504	0.670	0.508	0.544	0.705	0.582
30	1.000	0.995	0.853	0.501	0.668	0.503	0.540	0.701	0.574
31	1.000	1.000	0.879	0.500	0.667	0.500	0.532	0.695	0.561
32	1.000	1.000	0.884	0.500	0.667	0.500	0.531	0.693	0.558
33	1.000	1.000	0.884	0.500	0.667	0.500	0.531	0.693	0.558
34	1.000	1.000	0.905	0.500	0.667	0.500	0.525	0.688	0.547
35	1.000	1.000	0.921	0.500	0.667	0.500	0.521	0.685	0.539
36	1.000	1.000	0.942	0.500	0.667	0.500	0.515	0.680	0.529
37	1.000	1.000	0.942	0.500	0.667	0.500	0.515	0.680	0.529
38	1.000	1.000	0.958	0.500	0.667	0.500	0.511	0.676	0.521
39	1.000	1.000	0.968	0.500	0.667	0.500	0.508	0.674	0.516
40	1.000	1.000	0.984	0.500	0.667	0.500	0.504	0.670	0.508
41	1.000	1.000	0.989	0.500	0.667	0.500	0.503	0.669	0.505
42	1.000	1.000	0.989	0.500	0.667	0.500	0.503	0.669	0.505
43	1.000	1.000	0.995	0.500	0.667	0.500	0.501	0.668	0.503
44	1.000	1.000	0.995	0.500	0.667	0.500	0.501	0.668	0.503
45	1.000	1.000	1.000	0.500	0.667	0.500	0.500	0.667	0.500
46	1.000	1.000	1.000	0.500	0.667	0.500	0.500	0.667	0.500
47	1.000	1.000	1.000	0.500	0.667	0.500	0.500	0.667	0.500
48	1.000	1.000	1.000	0.500	0.667	0.500	0.500	0.667	0.500
49	1.000	1.000	1.000	0.500	0.667	0.500	0.500	0.667	0.500
50	1.000	1.000	1.000	0.500	0.667	0.500	0.500	0.667	0.500

A.1.4 Kinerja Manhattan Distance pada Dataset Privat

Berikut ini adalah tabel confusion matrix Manhattan Distance pada Dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	190	190	0	190	0
2	0	190	190	0	190	0
3	0	190	190	0	190	0
4	0	190	190	0	190	0
5	0	190	190	0	190	0
6	0	190	190	0	190	0
7	0	190	190	0	190	0
8	2	188	190	0	190	0
9	5	185	190	0	190	0
10	8	182	190	0	190	0
11	14	176	187	3	190	0
12	19	171	183	7	188	2
13	34	156	177	13	186	4
14	50	140	167	23	186	4
15	72	118	160	30	182	8
16	98	92	152	38	177	13
17	123	67	142	48	176	14
18	139	51	132	58	168	22
19	153	37	119	71	159	31
20	162	28	111	79	153	37
21	175	15	100	90	142	48
22	185	5	86	104	136	54
23	187	3	74	116	127	63
24	189	1	62	128	112	78
25	190	0	57	133	106	84
26	190	0	48	142	96	94
27	190	0	42	148	82	108
28	190	0	33	157	75	115
29	190	0	25	165	71	119
30	190	0	20	170	66	124
31	190	0	16	174	62	128
32	190	0	13	177	60	130
33	190	0	9	181	54	136
34	190	0	6	184	49	141
35	190	0	2	188	42	148
36	190	0	1	189	38	152
37	190	0	0	190	28	162
38	190	0	0	190	27	163
39	190	0	0	190	25	165
40	190	0	0	190	23	167
41	190	0	0	190	20	170
42	190	0	0	190	17	173
43	190	0	0	190	14	176
44	190	0	0	190	11	179
45	190	0	0	190	11	179
46	190	0	0	190	9	181
47	190	0	0	190	8	182
48	190	0	0	190	6	184
49	190	0	0	190	3	187
50	190	0	0	190	2	188

Berikut ini adalah tabel kinerja Manhattan pada dataset Privat

Thr	Recal	FPR-SF	FPR-RF	Precision-SF	F-Measure-SF	Acc-SF	Preci-RF	F-Meas-RF	Acc-RF
1	0.000	0.000	0.000	-	-	0.500	-	-	0.500
2	0.000	0.000	0.000	-	-	0.500	-	-	0.500
3	0.000	0.000	0.000	-	-	0.500	-	-	0.500
4	0.000	0.000	0.000	-	-	0.500	-	-	0.500
5	0.000	0.000	0.000	-	-	0.500	-	-	0.500
6	0.000	0.000	0.000	-	-	0.500	-	-	0.500
7	0.000	0.000	0.000	-	-	0.500	-	-	0.500
8	0.011	0.000	0.000	1.000	0.021	0.505	1.000	0.021	0.505
9	0.026	0.000	0.000	1.000	0.051	0.513	1.000	0.051	0.513
10	0.042	0.000	0.000	1.000	0.081	0.521	1.000	0.081	0.521
11	0.074	0.016	0.000	0.824	0.135	0.529	1.000	0.137	0.537
12	0.100	0.037	0.011	0.731	0.176	0.532	0.905	0.180	0.545
13	0.179	0.068	0.021	0.723	0.287	0.555	0.895	0.298	0.579
14	0.263	0.121	0.021	0.685	0.380	0.571	0.926	0.410	0.621
15	0.379	0.158	0.042	0.706	0.493	0.611	0.900	0.533	0.668
16	0.516	0.200	0.068	0.721	0.601	0.658	0.883	0.651	0.724
17	0.647	0.253	0.074	0.719	0.681	0.697	0.898	0.752	0.787
18	0.732	0.305	0.116	0.706	0.718	0.713	0.863	0.792	0.808
19	0.805	0.374	0.163	0.683	0.739	0.716	0.832	0.818	0.821
20	0.853	0.416	0.195	0.672	0.752	0.718	0.814	0.833	0.829
21	0.921	0.474	0.253	0.660	0.769	0.724	0.785	0.847	0.834
22	0.974	0.547	0.284	0.640	0.772	0.713	0.774	0.862	0.845
23	0.984	0.611	0.332	0.617	0.759	0.687	0.748	0.850	0.826
24	0.995	0.674	0.411	0.596	0.746	0.661	0.708	0.827	0.792
25	1.000	0.700	0.442	0.588	0.741	0.650	0.693	0.819	0.779
26	1.000	0.747	0.495	0.572	0.728	0.626	0.669	0.802	0.753
27	1.000	0.779	0.568	0.562	0.720	0.611	0.638	0.779	0.716
28	1.000	0.826	0.605	0.548	0.708	0.587	0.623	0.768	0.697
29	1.000	0.868	0.626	0.535	0.697	0.566	0.615	0.762	0.687
30	1.000	0.895	0.653	0.528	0.691	0.553	0.605	0.754	0.674
31	1.000	0.916	0.674	0.522	0.686	0.542	0.597	0.748	0.663
32	1.000	0.932	0.684	0.518	0.682	0.534	0.594	0.745	0.658
33	1.000	0.953	0.716	0.512	0.677	0.524	0.583	0.736	0.642
34	1.000	0.968	0.742	0.508	0.674	0.516	0.574	0.729	0.629
35	1.000	0.989	0.779	0.503	0.669	0.505	0.562	0.720	0.611
36	1.000	0.995	0.800	0.501	0.668	0.503	0.556	0.714	0.600
37	1.000	1.000	0.853	0.500	0.667	0.500	0.540	0.701	0.574
38	1.000	1.000	0.858	0.500	0.667	0.500	0.538	0.700	0.571
39	1.000	1.000	0.868	0.500	0.667	0.500	0.535	0.697	0.566
40	1.000	1.000	0.879	0.500	0.667	0.500	0.532	0.695	0.561
41	1.000	1.000	0.895	0.500	0.667	0.500	0.528	0.691	0.553
42	1.000	1.000	0.911	0.500	0.667	0.500	0.523	0.687	0.545
43	1.000	1.000	0.926	0.500	0.667	0.500	0.519	0.683	0.537
44	1.000	1.000	0.942	0.500	0.667	0.500	0.515	0.680	0.529
45	1.000	1.000	0.942	0.500	0.667	0.500	0.515	0.680	0.529
46	1.000	1.000	0.953	0.500	0.667	0.500	0.512	0.677	0.524
47	1.000	1.000	0.958	0.500	0.667	0.500	0.511	0.676	0.521
48	1.000	1.000	0.968	0.500	0.667	0.500	0.508	0.674	0.516
49	1.000	1.000	0.984	0.500	0.667	0.500	0.504	0.670	0.508
50	1.000	1.000	0.989	0.500	0.667	0.500	0.503	0.669	0.505

A.2 Perbandingan Kinerja Fitur Baru

A.2.1 Kinerja Dengan penambahan Fitur Baru pada Dataset SVC2004

Berikut adalah confusion matrix semua fitur baru pada Dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	400	800	0	800	0
2	0	400	800	0	800	0
3	0	400	800	0	800	0
4	0	400	800	0	800	0
5	0	400	800	0	800	0
6	0	400	800	0	800	0
7	0	400	800	0	800	0
8	5	395	800	0	800	0
9	9	391	799	1	800	0
10	25	375	796	4	800	0
11	49	351	785	15	800	0
12	84	316	775	25	799	1
13	123	277	756	44	798	2
14	167	233	726	74	791	9
15	209	191	678	122	782	18
16	250	150	635	165	769	31
17	298	102	592	208	751	49
18	334	66	551	249	736	64
19	350	50	499	301	724	76
20	364	36	434	366	707	93
21	382	18	374	426	682	118
22	392	8	314	486	659	141
23	396	4	253	547	628	172
24	397	3	211	589	598	202
25	397	3	172	628	562	238
26	398	2	142	658	527	273
27	398	2	114	686	482	318
28	398	2	91	709	452	348
29	398	2	75	725	418	382
30	399	1	65	735	390	410
31	400	0	57	743	349	451
32	400	0	45	755	323	477
33	400	0	33	767	290	510
34	400	0	27	773	273	527
35	400	0	21	779	245	555
36	400	0	16	784	213	587
37	400	0	11	789	193	607
38	400	0	9	791	175	625
39	400	0	4	796	154	646
40	400	0	3	797	135	665
41	400	0	3	797	116	684
42	400	0	3	797	105	695
43	400	0	3	797	93	707
44	400	0	1	799	78	722
45	400	0	0	800	65	735
46	400	0	0	800	53	747
47	400	0	0	800	46	754
48	400	0	0	800	42	758
49	400	0	0	800	37	763
50	400	0	0	800	33	767

Berikut ini adalah tabel kinerja Fitur baru pada dataset SVC2004

Thr	Recal	FPR-SF	FPR-RF	Precision-SF	F-Measure-SF	Acc-SF	Preci-RF	F-Meas-RF	Acc-RF
1	0.000	0.000	0.000	-	-	0.667	-	-	0.667
2	0.000	0.000	0.000	-	-	0.667	-	-	0.667
3	0.000	0.000	0.000	-	-	0.667	-	-	0.667
4	0.000	0.000	0.000	-	-	0.667	-	-	0.667
5	0.000	0.000	0.000	-	-	0.667	-	-	0.667
6	0.000	0.000	0.000	-	-	0.667	-	-	0.667
7	0.000	0.000	0.000	-	-	0.667	-	-	0.667
8	0.013	0.000	0.000	1.000	0.025	0.671	1.000	0.025	0.671
9	0.023	0.001	0.000	0.900	0.044	0.673	1.000	0.044	0.674
10	0.063	0.005	0.000	0.862	0.117	0.684	1.000	0.118	0.688
11	0.123	0.019	0.000	0.766	0.211	0.695	1.000	0.218	0.708
12	0.210	0.031	0.001	0.771	0.330	0.716	0.988	0.346	0.736
13	0.308	0.055	0.003	0.737	0.434	0.733	0.984	0.469	0.768
14	0.418	0.093	0.011	0.693	0.521	0.744	0.949	0.580	0.798
15	0.523	0.153	0.023	0.631	0.572	0.739	0.921	0.667	0.826
16	0.625	0.206	0.039	0.602	0.613	0.738	0.890	0.734	0.849
17	0.745	0.260	0.061	0.589	0.658	0.742	0.859	0.798	0.874
18	0.835	0.311	0.080	0.573	0.680	0.738	0.839	0.837	0.892
19	0.875	0.376	0.095	0.538	0.666	0.708	0.822	0.847	0.895
20	0.910	0.458	0.116	0.499	0.644	0.665	0.796	0.849	0.893
21	0.955	0.533	0.148	0.473	0.632	0.630	0.764	0.849	0.887
22	0.980	0.608	0.176	0.446	0.613	0.588	0.735	0.840	0.876
23	0.990	0.684	0.215	0.420	0.590	0.541	0.697	0.818	0.853
24	0.993	0.736	0.253	0.403	0.573	0.507	0.663	0.795	0.829
25	0.993	0.785	0.298	0.387	0.557	0.474	0.625	0.767	0.799
26	0.995	0.823	0.341	0.377	0.547	0.450	0.593	0.743	0.771
27	0.995	0.858	0.398	0.367	0.536	0.427	0.556	0.713	0.733
28	0.995	0.886	0.435	0.360	0.528	0.408	0.534	0.695	0.708
29	0.995	0.906	0.478	0.354	0.523	0.394	0.510	0.675	0.680
30	0.998	0.919	0.513	0.352	0.520	0.387	0.493	0.660	0.658
31	1.000	0.929	0.564	0.350	0.518	0.381	0.470	0.639	0.624
32	1.000	0.944	0.596	0.346	0.514	0.371	0.456	0.626	0.603
33	1.000	0.959	0.638	0.343	0.511	0.361	0.440	0.611	0.575
34	1.000	0.966	0.659	0.341	0.509	0.356	0.431	0.603	0.561
35	1.000	0.974	0.694	0.339	0.507	0.351	0.419	0.590	0.538
36	1.000	0.980	0.734	0.338	0.505	0.347	0.405	0.577	0.511
37	1.000	0.986	0.759	0.336	0.503	0.343	0.397	0.569	0.494
38	1.000	0.989	0.781	0.336	0.503	0.341	0.390	0.561	0.479
39	1.000	0.995	0.808	0.334	0.501	0.337	0.382	0.553	0.462
40	1.000	0.996	0.831	0.334	0.501	0.336	0.376	0.546	0.446
41	1.000	0.996	0.855	0.334	0.501	0.336	0.369	0.539	0.430
42	1.000	0.996	0.869	0.334	0.501	0.336	0.365	0.535	0.421
43	1.000	0.996	0.884	0.334	0.501	0.336	0.361	0.531	0.411
44	1.000	0.999	0.903	0.334	0.500	0.334	0.357	0.526	0.398
45	1.000	1.000	0.919	0.333	0.500	0.333	0.352	0.521	0.388
46	1.000	1.000	0.934	0.333	0.500	0.333	0.349	0.517	0.378
47	1.000	1.000	0.943	0.333	0.500	0.333	0.347	0.515	0.372
48	1.000	1.000	0.948	0.333	0.500	0.333	0.345	0.513	0.368
49	1.000	1.000	0.954	0.333	0.500	0.333	0.344	0.512	0.364
50	1.000	1.000	0.959	0.333	0.500	0.333	0.343	0.511	0.361

Berikut adalah confusion matrix 12 fitur baru terbaik di dataset SVC2004

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	400	800	0	800	0
2	0	400	800	0	800	0
3	0	400	800	0	800	0
4	0	400	800	0	800	0
5	0	400	800	0	800	0
6	0	400	800	0	800	0
7	1	399	800	0	800	0
8	5	395	800	0	800	0
9	13	387	800	0	800	0
10	23	377	794	6	800	0
11	46	354	783	17	800	0
12	82	318	775	25	800	0
13	124	276	752	48	799	1
14	164	236	724	76	789	11
15	221	179	677	123	776	24
16	263	137	635	165	758	42
17	312	88	578	222	747	53
18	343	57	537	263	721	79
19	364	36	474	326	698	102
20	378	22	407	393	673	127
21	391	9	358	442	643	157
22	397	3	294	506	624	176
23	397	3	235	565	596	204
24	398	2	188	612	556	244
25	399	1	159	641	507	293
26	399	1	128	672	462	338
27	399	1	107	693	404	396
28	399	1	86	714	360	440
29	399	1	73	727	307	493
30	400	0	55	745	272	528
31	400	0	40	760	239	561
32	400	0	26	774	211	589
33	400	0	20	780	179	621
34	400	0	12	788	155	645
35	400	0	8	792	138	662
36	400	0	6	794	119	681
37	400	0	3	797	109	691
38	400	0	1	799	99	701
39	400	0	1	799	85	715
40	400	0	1	799	69	731
41	400	0	0	800	56	744
42	400	0	0	800	39	761
43	400	0	0	800	30	770
44	400	0	0	800	21	779
45	400	0	0	800	13	787
46	400	0	0	800	9	791
47	400	0	0	800	6	794
48	400	0	0	800	3	797
49	400	0	0	800	2	798
50	400	0	0	800	0	800

Berikut adalah kinerja 12 fitur baru terbaik pada dataset SVC2004

Thr	Recall	FPR-SF	FPR-RF	Precision-SF	F-Measure-SF	Acc-SF	Preci-RF	F-Meas-RF	Acc-RF
1	0.000	0.000	0.000	-	-	0.667	-	-	0.667
2	0.000	0.000	0.000	-	-	0.667	-	-	0.667
3	0.000	0.000	0.000	-	-	0.667	-	-	0.667
4	0.000	0.000	0.000	-	-	0.667	-	-	0.667
5	0.000	0.000	0.000	-	-	0.667	-	-	0.667
6	0.000	0.000	0.000	-	-	0.667	-	-	0.667
7	0.003	0.000	0.000	1.000	0.005	0.668	1.000	0.005	0.668
8	0.013	0.000	0.000	1.000	0.025	0.671	1.000	0.025	0.671
9	0.033	0.000	0.000	1.000	0.063	0.678	1.000	0.063	0.678
10	0.058	0.008	0.000	0.793	0.107	0.681	1.000	0.109	0.686
11	0.115	0.021	0.000	0.730	0.199	0.691	1.000	0.206	0.705
12	0.205	0.031	0.000	0.766	0.323	0.714	1.000	0.340	0.735
13	0.310	0.060	0.001	0.721	0.434	0.730	0.992	0.472	0.769
14	0.410	0.095	0.014	0.683	0.513	0.740	0.937	0.570	0.794
15	0.553	0.154	0.030	0.642	0.594	0.748	0.902	0.685	0.831
16	0.658	0.206	0.053	0.614	0.635	0.748	0.862	0.746	0.851
17	0.780	0.278	0.066	0.584	0.668	0.742	0.855	0.816	0.883
18	0.858	0.329	0.099	0.566	0.682	0.733	0.813	0.835	0.887
19	0.910	0.408	0.128	0.528	0.668	0.698	0.781	0.841	0.885
20	0.945	0.491	0.159	0.490	0.646	0.654	0.749	0.835	0.876
21	0.978	0.553	0.196	0.469	0.634	0.624	0.714	0.825	0.862
22	0.993	0.633	0.220	0.440	0.609	0.576	0.693	0.816	0.851
23	0.993	0.706	0.255	0.413	0.583	0.527	0.661	0.793	0.828
24	0.995	0.765	0.305	0.394	0.565	0.488	0.620	0.764	0.795
25	0.998	0.801	0.366	0.384	0.554	0.465	0.577	0.731	0.755
26	0.998	0.840	0.423	0.373	0.542	0.439	0.541	0.702	0.718
27	0.998	0.866	0.495	0.365	0.535	0.422	0.502	0.668	0.669
28	0.998	0.893	0.550	0.358	0.527	0.404	0.476	0.644	0.633
29	0.998	0.909	0.616	0.354	0.523	0.393	0.447	0.618	0.588
30	1.000	0.931	0.660	0.349	0.518	0.379	0.431	0.602	0.560
31	1.000	0.950	0.701	0.345	0.513	0.367	0.416	0.588	0.533
32	1.000	0.968	0.736	0.341	0.508	0.355	0.404	0.576	0.509
33	1.000	0.975	0.776	0.339	0.506	0.350	0.392	0.563	0.483
34	1.000	0.985	0.806	0.337	0.504	0.343	0.383	0.554	0.463
35	1.000	0.990	0.828	0.336	0.503	0.340	0.377	0.547	0.448
36	1.000	0.993	0.851	0.335	0.502	0.338	0.370	0.540	0.433
37	1.000	0.996	0.864	0.334	0.501	0.336	0.367	0.537	0.424
38	1.000	0.999	0.876	0.334	0.500	0.334	0.363	0.533	0.416
39	1.000	0.999	0.894	0.334	0.500	0.334	0.359	0.528	0.404
40	1.000	0.999	0.914	0.334	0.500	0.334	0.354	0.523	0.391
41	1.000	1.000	0.930	0.333	0.500	0.333	0.350	0.518	0.380
42	1.000	1.000	0.951	0.333	0.500	0.333	0.345	0.512	0.366
43	1.000	1.000	0.963	0.333	0.500	0.333	0.342	0.510	0.358
44	1.000	1.000	0.974	0.333	0.500	0.333	0.339	0.507	0.351
45	1.000	1.000	0.984	0.333	0.500	0.333	0.337	0.504	0.344
46	1.000	1.000	0.989	0.333	0.500	0.333	0.336	0.503	0.341
47	1.000	1.000	0.993	0.333	0.500	0.333	0.335	0.502	0.338
48	1.000	1.000	0.996	0.333	0.500	0.333	0.334	0.501	0.336
49	1.000	1.000	0.998	0.333	0.500	0.333	0.334	0.501	0.335
50	1.000	1.000	1.000	0.333	0.500	0.333	0.333	0.500	0.333

A.2.2 Kinerja Dengan Penambahan Fitur Baru pada Dataset Privat

Berikut adalah confusion matrix semua fitur baru pada dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	190	190	0	190	0
2	0	190	190	0	190	0
3	0	190	190	0	190	0
4	0	190	190	0	190	0
5	0	190	190	0	190	0
6	0	190	190	0	190	0
7	0	190	190	0	190	0
8	0	190	190	0	190	0
9	4	186	190	0	190	0
10	8	182	190	0	190	0
11	14	176	190	0	190	0
12	19	171	186	4	190	0
13	33	157	182	8	189	1
14	49	141	169	21	187	3
15	74	116	161	29	186	4
16	103	87	155	35	185	5
17	125	65	149	41	179	11
18	143	47	141	49	178	12
19	155	35	131	59	171	19
20	167	23	117	73	162	28
21	174	16	99	91	153	37
22	184	6	84	106	136	54
23	188	2	72	118	127	63
24	189	1	63	127	120	70
25	190	0	51	139	109	81
26	190	0	45	145	99	91
27	190	0	38	152	93	97
28	190	0	34	156	83	107
29	190	0	28	162	77	113
30	190	0	23	167	74	116
31	190	0	18	172	67	123
32	190	0	14	176	61	129
33	190	0	11	179	59	131
34	190	0	8	182	53	137
35	190	0	5	185	49	141
36	190	0	4	186	45	145
37	190	0	2	188	38	152
38	190	0	1	189	34	156
39	190	0	0	190	30	160
40	190	0	0	190	25	165
41	190	0	0	190	21	169
42	190	0	0	190	17	173
43	190	0	0	190	15	175
44	190	0	0	190	13	177
45	190	0	0	190	12	178
46	190	0	0	190	11	179
47	190	0	0	190	8	182
48	190	0	0	190	7	183
49	190	0	0	190	6	184
50	190	0	0	190	6	184

Berikut ini adalah kinerja semua fitur baru pada dataset Privat

Thr	Recal	FPR-SF	FPR-RF	Precision-SF	F-Measure-SF	Acc-SF	Preci-RF	F-Meas-RF	Acc-RF
1	0.000	0.000	0.000	-	-	0.500	-	-	0.500
2	0.000	0.000	0.000	-	-	0.500	-	-	0.500
3	0.000	0.000	0.000	-	-	0.500	-	-	0.500
4	0.000	0.000	0.000	-	-	0.500	-	-	0.500
5	0.000	0.000	0.000	-	-	0.500	-	-	0.500
6	0.000	0.000	0.000	-	-	0.500	-	-	0.500
7	0.000	0.000	0.000	-	-	0.500	-	-	0.500
8	0.000	0.000	0.000	-	-	0.500	-	-	0.500
9	0.021	0.000	0.000	1.000	0.041	0.511	1.000	0.041	0.511
10	0.042	0.000	0.000	1.000	0.081	0.521	1.000	0.081	0.521
11	0.074	0.000	0.000	1.000	0.137	0.537	1.000	0.137	0.537
12	0.100	0.021	0.000	0.826	0.178	0.539	1.000	0.182	0.550
13	0.174	0.042	0.005	0.805	0.286	0.566	0.971	0.295	0.584
14	0.258	0.111	0.016	0.700	0.377	0.574	0.942	0.405	0.621
15	0.389	0.153	0.021	0.718	0.505	0.618	0.949	0.552	0.684
16	0.542	0.184	0.026	0.746	0.628	0.679	0.954	0.691	0.758
17	0.658	0.216	0.058	0.753	0.702	0.721	0.919	0.767	0.800
18	0.753	0.258	0.063	0.745	0.749	0.747	0.923	0.829	0.845
19	0.816	0.311	0.100	0.724	0.767	0.753	0.891	0.852	0.858
20	0.879	0.384	0.147	0.696	0.777	0.747	0.856	0.868	0.866
21	0.916	0.479	0.195	0.657	0.765	0.718	0.825	0.868	0.861
22	0.968	0.558	0.284	0.634	0.767	0.705	0.773	0.860	0.842
23	0.989	0.621	0.332	0.614	0.758	0.684	0.749	0.853	0.829
24	0.995	0.668	0.368	0.598	0.747	0.663	0.730	0.842	0.813
25	1.000	0.732	0.426	0.578	0.732	0.634	0.701	0.824	0.787
26	1.000	0.763	0.479	0.567	0.724	0.618	0.676	0.807	0.761
27	1.000	0.800	0.511	0.556	0.714	0.600	0.662	0.797	0.745
28	1.000	0.821	0.563	0.549	0.709	0.589	0.640	0.780	0.718
29	1.000	0.853	0.595	0.540	0.701	0.574	0.627	0.771	0.703
30	1.000	0.879	0.611	0.532	0.695	0.561	0.621	0.766	0.695
31	1.000	0.905	0.647	0.525	0.688	0.547	0.607	0.755	0.676
32	1.000	0.926	0.679	0.519	0.683	0.537	0.596	0.747	0.661
33	1.000	0.942	0.689	0.515	0.680	0.529	0.592	0.744	0.655
34	1.000	0.958	0.721	0.511	0.676	0.521	0.581	0.735	0.639
35	1.000	0.974	0.742	0.507	0.673	0.513	0.574	0.729	0.629
36	1.000	0.979	0.763	0.505	0.671	0.511	0.567	0.724	0.618
37	1.000	0.989	0.800	0.503	0.669	0.505	0.556	0.714	0.600
38	1.000	0.995	0.821	0.501	0.668	0.503	0.549	0.709	0.589
39	1.000	1.000	0.842	0.500	0.667	0.500	0.543	0.704	0.579
40	1.000	1.000	0.868	0.500	0.667	0.500	0.535	0.697	0.566
41	1.000	1.000	0.889	0.500	0.667	0.500	0.529	0.692	0.555
42	1.000	1.000	0.911	0.500	0.667	0.500	0.523	0.687	0.545
43	1.000	1.000	0.921	0.500	0.667	0.500	0.521	0.685	0.539
44	1.000	1.000	0.932	0.500	0.667	0.500	0.518	0.682	0.534
45	1.000	1.000	0.937	0.500	0.667	0.500	0.516	0.681	0.532
46	1.000	1.000	0.942	0.500	0.667	0.500	0.515	0.680	0.529
47	1.000	1.000	0.958	0.500	0.667	0.500	0.511	0.676	0.521
48	1.000	1.000	0.963	0.500	0.667	0.500	0.509	0.675	0.518
49	1.000	1.000	0.968	0.500	0.667	0.500	0.508	0.674	0.516
50	1.000	1.000	0.968	0.500	0.667	0.500	0.508	0.674	0.516

Berikut adalah confusion matrix 12 fitur baru terbaik pada dataset Privat

Thr	TP	FN	TN-SF	FP-SF	TN-RF	FP-RF
1	0	190	190	0	190	0
2	0	190	190	0	190	0
3	0	190	190	0	190	0
4	0	190	190	0	190	0
5	0	190	190	0	190	0
6	0	190	190	0	190	0
7	0	190	190	0	190	0
8	0	190	190	0	190	0
9	4	186	190	0	190	0
10	7	183	190	0	190	0
11	15	175	190	0	190	0
12	19	171	186	4	190	0
13	33	157	181	9	187	3
14	52	138	169	21	184	6
15	75	115	161	29	180	10
16	101	89	155	35	180	10
17	128	62	149	41	179	11
18	146	44	140	50	177	13
19	156	34	131	59	170	20
20	167	23	116	74	160	30
21	176	14	102	88	152	38
22	185	5	85	105	140	50
23	188	2	71	119	128	62
24	189	1	61	129	113	77
25	190	0	48	142	92	98
26	190	0	42	148	79	111
27	190	0	37	153	63	127
28	190	0	34	156	56	134
29	190	0	25	165	47	143
30	190	0	23	167	44	146
31	190	0	16	174	42	148
32	190	0	11	179	40	150
33	190	0	10	180	38	152
34	190	0	7	183	33	157
35	190	0	5	185	29	161
36	190	0	3	187	25	165
37	190	0	2	188	21	169
38	190	0	1	189	14	176
39	190	0	0	190	12	178
40	190	0	0	190	12	178
41	190	0	0	190	11	179
42	190	0	0	190	10	180
43	190	0	0	190	9	181
44	190	0	0	190	7	183
45	190	0	0	190	7	183
46	190	0	0	190	6	184
47	190	0	0	190	5	185
48	190	0	0	190	2	188
49	190	0	0	190	2	188
50	190	0	0	190	0	190

Berikut adalah kinerja 12 fitur terbaik pada dataset Privat

Thr	Recal	FPR-SF	FPR-RF	Precision-SF	F-Measure-SF	Acc-SF	Preci-RF	F-Meas-RF	Acc-RF
1	0.000	0.000	0.000	-	-	0.500	-	-	0.500
2	0.000	0.000	0.000	-	-	0.500	-	-	0.500
3	0.000	0.000	0.000	-	-	0.500	-	-	0.500
4	0.000	0.000	0.000	-	-	0.500	-	-	0.500
5	0.000	0.000	0.000	-	-	0.500	-	-	0.500
6	0.000	0.000	0.000	-	-	0.500	-	-	0.500
7	0.000	0.000	0.000	-	-	0.500	-	-	0.500
8	0.000	0.000	0.000	-	-	0.500	-	-	0.500
9	0.021	0.000	0.000	1.000	0.041	0.511	1.000	0.041	0.511
10	0.037	0.000	0.000	1.000	0.071	0.518	1.000	0.071	0.518
11	0.079	0.000	0.000	1.000	0.146	0.539	1.000	0.146	0.539
12	0.100	0.021	0.000	0.826	0.178	0.539	1.000	0.182	0.550
13	0.174	0.047	0.016	0.786	0.284	0.563	0.917	0.292	0.579
14	0.274	0.111	0.032	0.712	0.395	0.582	0.897	0.419	0.621
15	0.395	0.153	0.053	0.721	0.510	0.621	0.882	0.545	0.671
16	0.532	0.184	0.053	0.743	0.620	0.674	0.910	0.671	0.739
17	0.674	0.216	0.058	0.757	0.713	0.729	0.921	0.778	0.808
18	0.768	0.263	0.068	0.745	0.756	0.753	0.918	0.837	0.850
19	0.821	0.311	0.105	0.726	0.770	0.755	0.886	0.852	0.858
20	0.879	0.389	0.158	0.693	0.775	0.745	0.848	0.863	0.861
21	0.926	0.463	0.200	0.667	0.775	0.732	0.822	0.871	0.863
22	0.974	0.553	0.263	0.638	0.771	0.711	0.787	0.871	0.855
23	0.989	0.626	0.326	0.612	0.757	0.682	0.752	0.855	0.832
24	0.995	0.679	0.405	0.594	0.744	0.658	0.711	0.829	0.795
25	1.000	0.747	0.516	0.572	0.728	0.626	0.660	0.795	0.742
26	1.000	0.779	0.584	0.562	0.720	0.611	0.631	0.774	0.708
27	1.000	0.805	0.668	0.554	0.713	0.597	0.599	0.750	0.666
28	1.000	0.821	0.705	0.549	0.709	0.589	0.586	0.739	0.647
29	1.000	0.868	0.753	0.535	0.697	0.566	0.571	0.727	0.624
30	1.000	0.879	0.768	0.532	0.695	0.561	0.565	0.722	0.616
31	1.000	0.916	0.779	0.522	0.686	0.542	0.562	0.720	0.611
32	1.000	0.942	0.789	0.515	0.680	0.529	0.559	0.717	0.605
33	1.000	0.947	0.800	0.514	0.679	0.526	0.556	0.714	0.600
34	1.000	0.963	0.826	0.509	0.675	0.518	0.548	0.708	0.587
35	1.000	0.974	0.847	0.507	0.673	0.513	0.541	0.702	0.576
36	1.000	0.984	0.868	0.504	0.670	0.508	0.535	0.697	0.566
37	1.000	0.989	0.889	0.503	0.669	0.505	0.529	0.692	0.555
38	1.000	0.995	0.926	0.501	0.668	0.503	0.519	0.683	0.537
39	1.000	1.000	0.937	0.500	0.667	0.500	0.516	0.681	0.532
40	1.000	1.000	0.937	0.500	0.667	0.500	0.516	0.681	0.532
41	1.000	1.000	0.942	0.500	0.667	0.500	0.515	0.680	0.529
42	1.000	1.000	0.947	0.500	0.667	0.500	0.514	0.679	0.526
43	1.000	1.000	0.953	0.500	0.667	0.500	0.512	0.677	0.524
44	1.000	1.000	0.963	0.500	0.667	0.500	0.509	0.675	0.518
45	1.000	1.000	0.963	0.500	0.667	0.500	0.509	0.675	0.518
46	1.000	1.000	0.968	0.500	0.667	0.500	0.508	0.674	0.516
47	1.000	1.000	0.974	0.500	0.667	0.500	0.507	0.673	0.513
48	1.000	1.000	0.989	0.500	0.667	0.500	0.503	0.669	0.505
49	1.000	1.000	0.989	0.500	0.667	0.500	0.503	0.669	0.505
50	1.000	1.000	1.000	0.500	0.667	0.500	0.500	0.667	0.500

(lembar ini sengaja dikosongkan)

BIOGRAFI PENULIS



Ano Ranga Rahardika. Lahir di Madiun, 30 Nopember 1990, anak kedua dari dua bersaudara. Penulis menempuh pendidikan formal mulai dari tahun 1997-2003 di SD N 01 Bangunsari Madiun, 2003-2006 di SMP N 1 Dolopo, 2006-2009 di SMA N 3 Madiun. Selanjutnya pada tahun 2013 penulis menyelesaikan pendidikan Strata Satu di Institut Teknologi Telkom Bandung (IT TELKOM) jurusan Teknik Informatika. Tahun 2013 penulis mulai bekerja di PT ARM SOLUSI sebagai *software engineer* yang menangani Notadinas Elektronik diantaranya di PT PJB dan PT Telkom Indonesia sampai hari ini. Tahun 2014 penulis diterima sebagai mahasiswa Program Pascasarjana Institut Teknologi Sepuluh Nopember Surabaya, Fakultas Teknologi Informasi dan Komunikasi, Departemen Sistem Informasi dengan NRP. 05211450010018 . *E-mail:* anorangga@gmail.com.