

BAB IV

HASIL DAN ANALISIS

4.1 Pendahuluan

Pada bab ini akan dibahas hasil dan analisis dari hasil penelitian yang telah dilakukan. Pada tahapan *offline* telah dilakukan perhitungan matematis dan hasil dari tersebut akan di uji secara *real time* pada tahapan *online*. Tahapan *offline* ini hanya menggunakan satu lantai, yaitu lantai 1 dari total tiga lantai. Total ruangan dari lantai 1 Gedung D Fasilkom berjumlah empat ruangan, yaitu Ruang A.1, Ruang B.1, Ruang C.1, dan Ruang D.1.

4.2 Tahap Offline

4.2.1 Radio Map atau Database

Tahapan *offline* dilakukan pengambilan data yang kemudian dibentuk *database*. *Database* berasal dari nilai data dari titik referensi sebanyak 86 titik yang telah ditentukan dan menggunakan 6 *Access Point* di gedung D Fakultas Ilmu Komputer, Universitas Sriwijaya. Dari 86 titik referensi dan 6 *access point* sebanyak 5.990 data yang digunakan.

20 titik referensi yang telah ditentukan terkumpul dari 4 kelas, dimana 1 kelas terdiri dari 5 titik referensi di dalam kelas dan 2 titik referensi di luar kelas.

Tabel 4.1 Kalkulasi Jumlah data yang diambil

Lokasi	Jumlah Titik Referensi	Jumlah Data RSS Access Point yang di dapat oleh tools	Jumlah data (RSS) access point yang digunakan
Ruang A.1	5 titik	22.898	1.520
Ruang B.1	5 titik	15.026	1.495
Ruang C.1	5 titik	47.024	1.505
Ruang D.1	5 titik	20.899	1470

4.2.2 Preprocessing Data

Preprocessing Data dilakukan untuk membuat data yang telah didapatkan dari *access point* diperbaiki presisi dan kinerja data agar menjadi sebuah data yang siap digunakan sebagai data *training* maupun data *testing*. Berikut langkah-langkah *preprocessing* data :

4.2.2.1 Pemilihan Fitur dan Label Data

Strategi pertama pada *preprocessing* data adalah pemilihan fitur dan label data yang dibangun yang bertujuan menyederhanakan data mentah yang ada. *Access Point* dan label data yang telah di lakukan dapat dilihat pada tabel 3 berikut :

Tabel 4.2 Fitur Data Fasilkom

Access Points (APs)	Fitur
00:18:f8:4a:57:cb	AP1
10:fe:ed:bc:18:bd	AP2
06:18:d6:e7:7a:89	AP3
46:d9:e7:fe:82:2c	AP4
26:18:d6:e7:7a:89	AP5

Tabel 4.3 Label Data Fasilkom

Lokasi Referensi Point	Label
Ruang A.1	0
Ruang B.1	1
Ruang C.1	2
Ruang D.1	3

4.1.1.1 Normalisasi Data

Tahapan *preprocessing data* selanjutnya adalah normalisasi data. Normalisasi data digunakan untuk memodifikasi nilai dalam variabel sehingga dapat mengukur data ke dalam skala umum. Pada penelitian ini digunakan berbagai bentuk normalisasi. Tujuan yang paling umum normalisasi adalah untuk mengubah nilai-nilai sehingga *range* nilai nya berada diantara 0 dan 1.

Tahapan ini makan menyesuaikan rentang nilai yang dapat digunakan pada fungsi aktivasi. Sehingga normalisasi data bertujuan agar mendapatkan fungsi aktivasi yang sesuai untuk digunakan. Pada penelitian ini menggunakan normalisasi MinMax dengan penghitungan pada persamaan (3.1). bertujuan agar mendapatkan fungsi aktivasi yang sesuai, normalisasi dilakukan karena pada *database* nilai RSS berupa nilai negatif. Berikut normalisasi yang telah dilakukan menggunakan teknik MinMax.

Index	AP1	AP2	AP3	AP4	AP5	Label
0	-100	-78	-70	-100	-100	0
1	-100	-100	-67	-100	-100	0
2	-100	-100	-66	-100	-100	0
3	-100	-76	-68	-100	-100	0
4	-100	-76	-69	-100	-100	0
5	-100	-100	-68	-100	-100	0
6	-100	-75	-73	-100	-100	0
7	-100	-75	-70	-100	-100	0
8	-100	-77	-69	-100	-100	0
9	-100	-100	-67	-100	-100	0
10	-100	-76	-70	-100	-100	0
11	-100	-78	-67	-100	-100	0
12	-100	-100	-69	-100	-100	0
13	-100	-100	-68	-100	-100	0
14	-100	-74	-67	-100	-100	0
15	-100	-79	-71	-100	-100	0
16	-100	-100	-68	-100	-100	0

Gambar 4.1 Capture Data RSS sebelum Normalisasi

	0	1	2	3	4
0	0	0.127168	0.459364	0	0
1	0	0	0.469965	0	0
2	0	0	0.473498	0	0
3	0	0.138728	0.466431	0	0
4	0	0.138728	0.462898	0	0
5	0	0	0.466431	0	0
6	0	0.144509	0.448763	0	0
7	0	0.144509	0.459364	0	0
8	0	0.132948	0.462898	0	0
9	0	0	0.469965	0	0
10	0	0.138728	0.459364	0	0
11	0	0.127168	0.469965	0	0
12	0	0	0.462898	0	0
13	0	0	0.466431	0	0
14	0	0.150289	0.469965	0	0
15	0	0.121387	0.45583	0	0
16	0	0	0.466431	0	0

Gambar 4.2 Capture Data fitur Awal Normalisasi

	0	1	2	3	4
0	-0.571248	0.985565	2.27337	-0.53972	-0.432211
1	-0.571248	-0.60678	2.55726	-0.53972	-0.432211
2	-0.571248	-0.60678	2.65189	-0.53972	-0.432211
3	-0.571248	1.13032	2.46263	-0.53972	-0.432211
4	-0.571248	1.13032	2.368	-0.53972	-0.432211
5	-0.571248	-0.60678	2.46263	-0.53972	-0.432211
6	-0.571248	1.2027	1.98947	-0.53972	-0.432211
7	-0.571248	1.2027	2.27337	-0.53972	-0.432211
8	-0.571248	1.05794	2.368	-0.53972	-0.432211
9	-0.571248	-0.60678	2.55726	-0.53972	-0.432211
10	-0.571248	1.13032	2.27337	-0.53972	-0.432211
11	-0.571248	0.985565	2.55726	-0.53972	-0.432211
12	-0.571248	-0.60678	2.368	-0.53972	-0.432211
13	-0.571248	-0.60678	2.46263	-0.53972	-0.432211
14	-0.571248	1.27508	2.55726	-0.53972	-0.432211
15	-0.571248	0.913185	2.17874	-0.53972	-0.432211
16	-0.571248	-0.60678	2.46263	-0.53972	-0.432211

Gambar 4.3 Capture Data fitur Normalisasi Min-Max

4.3 Algoritma Metode *Support Vectorn Machine* (SVM)

Metode analisis yang akan digunakan dalam penelitian adalah dengan menggunakan metode SVM. Pada penggunaan metode SVM akan dilakukan pemisahan antara *training dataset* dengan *testing dataset*. Desain program ini terdapat di lampiran. *Training dataset* diambil 80% keseluruhan dari 1880 data sampel. Sedangkan untuk *testing dataset* merupakan sisa dari data yang akan diuji akurasi. Pengambilan *training dataset* maupun *testing dataset* dilakukan secara acak dengan bantuan pemrograman *python*. Seperti yang telah dijelaskan pada metode penelitian dalam bab sebelumnya, akan menggunakan fungsi linear SVM untuk mencari nilai akurasi terbaik.

4.2.1 RBF Kernel SVM

RBF Kernel merupakan fungsi kernel yang digunakan ketika data tidak terpisah secara linear. Dalam melakukan analisis dengan fungsi RBF kernel, dilakukan optimasi parameter *Cost* (C) dan *Gamma* (γ). Analisis dilakukan dengan menggunakan 1880 sampel yang dibagi menjadi dua bagian yaitu 80% sebagai training dataset dan 20% lainnya sebagai testing dataset. Sama halnya seperti pada fungsi linear kernel, dalam penentuan parameter terbaik dalam RBF kernel juga dilakukan *trial and error* sehingga akan diperoleh tabel berikut.

Tabel 4.4 Nilai akurasi parameter terbaik untuk *training dataset*

Parameter	Akurasi				
	$\gamma = 1$	$\gamma = 2$	$\gamma = 3$	$\gamma = 4$	$\gamma = 5$
$C = 0.01$	0.3952	0.5033	0.5033	0.5033	0.5033
$C = 0.1$	0.5608	0.5855	0.6013	0.6396	0.6734
$C = 1$	0.7229	0.7297	0.7297	0.7286	0.7680
$C = 10$	0.7274	0.7668	0.7939	0.8006	0.7984
$C = 100$	0.7972	0.8006	0.8006	0.8063	0.8040

Tabel 4.5 Nilai akurasi parameter terbaik untuk *testing dataset*

Parameter	Akurasi				
	$\gamma = 1$	$\gamma = 2$	$\gamma = 3$	$\gamma = 4$	$\gamma = 5$
$C = 0.01$	0.3310	0.4391	0.4425	0.4459	0.4459
$C = 0.1$	0.5135	0.5371	0.5608	0.5912	0.6283
$C = 1$	0.6925	0.7128	0.7128	0.7128	0.7263
$C = 10$	0.7128	0.7263	0.7837	0.7871	0.7668
$C = 100$	0.7601	0.7668	0.7668	0.7871	0.7668

Penentuan parameter terbaik dapat dilakukan dengan memilih salah satu nilai parameter *cost* dan γ . Penulis akan menggunakan $C = 100$, serta $\gamma = 4$ sebagai parameter terbaik dalam pembentukan model dengan *training dataset*. Sehingga akan diperoleh parameter model adalah sebagai berikut.

4.2.2 Linear Kernel SVM

Linear Kernel SVM merupakan fungsi kernel yang baik digunakan ketika data sudah terpisah secara linear. Analisis dilakukan dengan menggunakan 1880 sampel yang dibagi menjadi dua bagian yaitu 80% sebagai *training dataset* dan 20% lainnya sebagai *testing dataset*. Dalam melakukan analisis dengan fungsi linear kernel, dilakukan optimasi parameter C atau *Cost*. Pengoptimalan parameter C dapat dilakukan dengan cara *trial and error*. Berikut ini merupakan parameter linear kernel terbaik dengan cara *trian and error*.

Tabel 4.6 Nilai akurasi parameter terbaik Linear kernel

Parameter	Akurasi	
	<i>Training Dataset</i>	<i>Testing Dataset</i>
C = 0.01	0.2590	0.2567
C = 0.1	0.5225	0.4527
C = 1	0.6700	0.6216
C = 10	0.7173	0.6993
C = 100	0.7308	0.7229

4.3.1 Polynomial Kernel SVM

Polinomial kernel merupakan fungsi kernel non linear yang sangat cocok digunakan untuk permasalahan yang semua *training dataset*-nya dinormalisasi. Parameter dari fungsi polinomial kernel terdiri dari *Cost* dan *Degree (d)*. Sama seperti halnya dalam fungsi –fungsi kernel sebelumnya, pengoptimalan parameter dapat dilakukan dengan cara *trial and error* sehingga akan didapatkan hasil seperti berikut.

Tabel 4.7 Nilai akurasi parameter terbaik untuk *training dataset*

Parameter	Akurasi	
	d = 1	d = 2
C = 0.01	0.2567	0.2567
C = 0.1	0.4797	0.2567
C = 1	0.5563	0.4380
C = 10	0.7038	0.5529
C = 100	0.7252	0.6069

Tabel 4.8 Nilai akurasi parameter terbaik untuk *testing dataset*

Parameter	Akurasi	
	d = 1	d = 2
C = 0.01	0.2567	0.2567
C = 0.1	0.4155	0.2567
C = 1	0.4662	0.4020
C = 10	0.6722	0.4560
C = 100	0.7027	0.5405

Berdasarkan dari beberapa metode SVM yang telah di uji coba, metode SVM RBF kernel merupakan metode dengan akurasi terbaik dengan akurasi *training dataset* pada $C = 100$ dan $\gamma = 4$., Untuk *testing dataset* didapatkan akurasi terbaik pada $C = 100$ dan $\gamma = 4$ sehingga langkah selanjutnya adalah dapat dibuat *confusion matrix* dari hasil metode SVM RBF kernel.

Tabel 4.9 Tabel *Confusion Matrix training dataset SVM RBF Kernel*

Prediksi	Aktual			
	0	1	2	3
0	189	39	0	0
1	6	193	25	0
2	0	23	184	8
3	1	67	5	148

Pada tabel klasifier diatas terdapat total 888 prediksi yang diperoleh dengan mengalisis *training dataset* untuk melihat kesesuaian pengambilan data.

Tabel 4.10 Tabel *Confusion Matrix* testing dataset SVM RBF Kernel

Prediksi	Aktual			
	0	1	2	3
0	55	21	0	0
1	2	58	14	1
2	0	8	63	1
3	0	18	4	51

Setelah diperoleh *confusion matrix* seperti diatas, dapat dilanjutkan dengan mencari nilai-nilai *accuracy*, *sensitivity*, *specificity*, *precision* dari *training dataset* dan *testing dataset* menggunakan perhitungan seperti berikut.

Tabel 4.11 Hasil Evaluasi *training* data untuk masing-masing kelas

Label	Acc	Sen	Spe	Pre
0	93%	82%	98%	96%
1	81%	86%	80%	59%
2	92%	85%	94%	85%
3	89%	67%	98%	95%
Rata-rata	89%	80%	92%	83%

Tabel 4.12 Hasil Evaluasi *testing* data untuk masing-masing kelas

Label	Acc	Sen	Spe	Pre
0	90%	72%	99%	94%
1	80%	89%	78%	55%
2	90%	87%	91%	78%
3	91%	70%	99%	96%
Rata-rata	88%	80%	92%	81%

SVM dilatih dengan menggunakan *training dataset*, kemudian performanya dievaluasikan ke dalam *testing dataset*. Ketika dilakukan analisis SVM dengan menggunakan fungsi RBF kernel, hasil menunjukkan bahwa SVM akan mengklasifikasikan secara benar 55 untuk label 0, 58 untuk label 1, 63 untuk label 2 dan 51 untuk label 3 dari 296 sampel *testing dataset*.

4.4 Tahapan Online

4.4.1 Preprocessing Data

Pada tahapan online ini akan dilakukan estimasi posisi dengan 3 lantai sekaligus. Pada tahapan ini akan dilakukan kembali *preprocessing data* karena di *online* estimasi posisi menggunakan 3 lantai. Pada tahapan ini akan dilakukan pemilihan fitur dan label data yang dibangun bertujuan menyederhanakan data mentah yang ada. *Access point* dan label data yang telah dilakukan dapat dilihat pada tabel berikut:

Tabel 4.13 Fitur Data Gedung D Fasilkom

Access Points (APs)	Fitur
38:17:c3:e7:fa:a0	AP1
38:17:c3:e7:fb:40	AP2
38:17:c3:e7:43:40	AP3
b0:b8:67:eb:09:e0	AP4
10:fe:ed:bc:18:bd	AP5
26:18:d6:e7:73:ed	AP6
06:18:d6:e7:73:ed	AP7
38:17:c3:e7:40:40	AP8
46:d9:e7:fe:82:2c	AP9
00:18:f8:4a:57:cb	AP10

Tabel 4.14 Label Data Gedung D Fasilkom

Lokasi Referensi Point	Label
Ruang A.1	0
Ruang B.1	1
Ruang C.1	2
Ruang D.1	3
Tangga 1	4
Ruang A.2	5
Ruang B.2	6
Ruang C.2	7
Ruang D.2	8
Tangga 2	9
Ruang A.3	10
Ruang B.3	11
Ruang C.3	12
Ruang D.3	13

Setelah diketahui fitur dan label data maka tahapan selanjutnya adalah normalisasi data untuk menyesuaikan rentang nilai yang dapat digunakan pada fungsi aktivasi yang sesuai untuk digunakan.

0	0	1	2	3	4	5	6	7	8	9
0	-71	-54	-67	-100	-71	-100	-100	-100	-100	-99
1	-71	-54	-67	-100	-71	-100	-100	-100	-100	-99
2	-67	-53	-66	-100	-77	-100	-100	-100	-100	-99
3	-67	-53	-66	-100	-77	-100	-100	-100	-100	-99
4	-66	-53	-65	-100	-84	-100	-100	-100	-100	-99
5	-66	-53	-65	-100	-84	-100	-100	-100	-100	-99
6	-67	-53	-100	-100	-100	-100	-100	-100	-100	-99
7	-67	-53	-100	-100	-100	-100	-100	-100	-100	-99
8	-74	-56	-100	-100	-75	-100	-100	-100	-100	-99
9	-74	-56	-100	-100	-75	-100	-100	-100	-100	-99
10	-68	-61	-70	-100	-100	-100	-100	-100	-100	-99
11	-68	-61	-70	-100	-100	-100	-100	-100	-100	-99
12	-67	-62	-69	-100	-72	-100	-100	-100	-100	-99
13	-67	-62	-69	-100	-72	-100	-100	-100	-100	-99
14	-100	-59	-70	-100	-100	-100	-100	-100	-100	-99
15	-100	-59	-70	-100	-100	-100	-100	-100	-100	-99
16	-67	-60	-69	-100	-73	-100	-100	-100	-100	-99

Gambar 4.4 Capture Fitur Data Sebelum normalisasi

	0	1	2	3	4	5	6	7	8	9
0	0.439394	0.71875	0.478261	0	0.783784	0	0	0	0	0.0555556
1	0.439394	0.71875	0.478261	0	0.783784	0	0	0	0	0.0555556
2	0.5	0.734375	0.492754	0	0.621622	0	0	0	0	0.0555556
3	0.5	0.734375	0.492754	0	0.621622	0	0	0	0	0.0555556
4	0.515152	0.734375	0.507246	0	0.432432	0	0	0	0	0.0555556
5	0.515152	0.734375	0.507246	0	0.432432	0	0	0	0	0.0555556
6	0.5	0.734375	0	0	0	0	0	0	0	0.0555556
7	0.5	0.734375	0	0	0	0	0	0	0	0.0555556
8	0.393939	0.6875	0	0	0.675676	0	0	0	0	0.0555556
9	0.393939	0.6875	0	0	0.675676	0	0	0	0	0.0555556
10	0.484848	0.609375	0.434783	0	0	0	0	0	0	0.0555556
11	0.484848	0.609375	0.434783	0	0	0	0	0	0	0.0555556
12	0.5	0.59375	0.449275	0	0.756757	0	0	0	0	0.0555556
13	0.5	0.59375	0.449275	0	0.756757	0	0	0	0	0.0555556
14	0	0.640625	0.434783	0	0	0	0	0	0	0.0555556
15	0	0.640625	0.434783	0	0	0	0	0	0	0.0555556
16	0.5	0.625	0.449275	0	0.72973	0	0	0	0	0.0555556

Gambar 4.5 Capture Data fitur MinMax

0	0.439597	1.52176	0.510714	-0.527317	2.14253	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
1	0.439597	1.52176	0.510714	-0.527317	2.14253	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
2	0.657737	1.57425	0.567514	-0.527317	1.58192	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
3	0.657737	1.57425	0.567514	-0.527317	1.58192	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
4	0.712272	1.57425	0.624314	-0.527317	0.927872	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
5	0.712272	1.57425	0.624314	-0.527317	0.927872	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
6	0.657737	1.57425	-1.36369	-0.527317	-0.567095	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
7	0.657737	1.57425	-1.36369	-0.527317	-0.567095	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
8	0.275991	1.41678	-1.36369	-0.527317	1.76879	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
9	0.275991	1.41678	-1.36369	-0.527317	1.76879	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
10	0.603202	1.15433	0.340313	-0.527317	-0.567095	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
11	0.603202	1.15433	0.340313	-0.527317	-0.567095	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
12	0.657737	1.10184	0.397113	-0.527317	2.0491	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
13	0.657737	1.10184	0.397113	-0.527317	2.0491	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
14	-1.14192	1.25931	0.340313	-0.527317	-0.567095	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
15	-1.14192	1.25931	0.340313	-0.527317	-0.567095	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746
16	0.657737	1.20682	0.397113	-0.527317	1.95566	-0.164513	-0.147326	-0.59719	-0.240013	-0.433746

Gambar 4.6 Capture Data Fitur Normalisasi

4.4.2 Klasifikasi Menggunakan *Support Vector Machine* (SVM)

Pada tahapan ini sama seperti percobaan lantai 1 sebelumnya setelah nilai fitur yang di normalisasi akan diklasifikasi sesuai dengan kelas masing-masing dengan menggunakan metode SVM kernel.

Dalam melakukan klasifikasi dengan kernel dilakukan dengan melakukan *trial and error* dengan kernel RBF, *polynomial*, dan linear. Dari percobaan tersebut didapatkan hasil akurasi terbaik dari SVM Kernel.

Tabel 4.15 Nilai akurasi parameter terbaik untuk *training dataset*

No.	Kernel	Akurasi
1	RBF	95.09%
2	Polynomial	88.92%
3	Linear	79.56%

Berdasarkan pada hasil tabel diatas, maka kernel dengan akurasi terbaik adalah kernel RBF maka setelah didapat kernel terbaik dapat lakukan evaluasi performa menggunakan *confusion matrix*.

Tabel 4.16 Tabel *Confusion Matrix Training Dataset*

Prediksi	Aktual													
	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	368	0	0	0	0	0	1	0	0	0	5	0	0	0
1	0	336	3	0	0	0	0	0	0	0	0	0	0	0
2	0	5	362	0	0	0	0	0	0	0	0	2	0	0
3	0	0	49	306	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	171	0	0	0	0	4	0	2	0	0
5	0	0	0	0	0	369	3	0	0	0	0	0	0	0
6	0	0	0	0	0	0	364	0	0	0	2	0	0	0
7	0	0	0	0	0	0	4	353	0	2	2	2	0	0
8	0	0	0	0	0	0	0	3	361	0	0	0	0	0
9	0	0	0	0	4	0	3	4	0	162	0	1	0	4
10	0	0	0	0	0	0	7	0	0	0	362	5	0	0
11	0	0	0	0	0	0	15	2	0	0	6	332	0	1
12	0	0	0	0	0	0	0	0	0	0	0	0	351	21
13	0	0	0	0	0	0	2	10	0	2	0	16	39	285

Dari tabel *confusion matrix* dapat diketahui nilai TP, TN, FP, dan FN yang mana nilai tersebut dapat mengukur kinerja dari model yang akan digunakan berupa nilai *sensitivity*, *specificity*, *precision*, dan *accuracy*.

Tabel 4.17 Hasil Evaluasi *training dataset*

Label	Acc	Sen	Spe	Pre
0	99%	98%	99%	98%
1	99%	99%	99%	97%
2	98%	95%	98%	85%
3	98%	84%	99%	98%
4	99%	89%	99%	89%
5	99%	97%	100%	100%
6	98%	100%	98%	86%
7	99%	100%	99%	96%
8	99%	97%	99%	98%
9	99%	83%	99%	92%
10	98%	91%	99%	91%
11	97%	84%	98%	86%
12	98%	91%	98%	85%
13	97%	72%	99%	87%
Rata-rata	99%	97%	99%	99%

4.5 Implementasi Estimasi Posisi Objek *Real-time*

Pada tahapan ini akan dilakukan implementasi estimasi posisi objek menggunakan metode fingerprint berdasarkan support vector machine (SVM). Pada prosesnya akan diambil 1 sampel per label dimana masing-masing ruangan terdiri dari 4 sampel uji. Hasil dan implementasinya dapat dilihat pada *lampiran 3*.

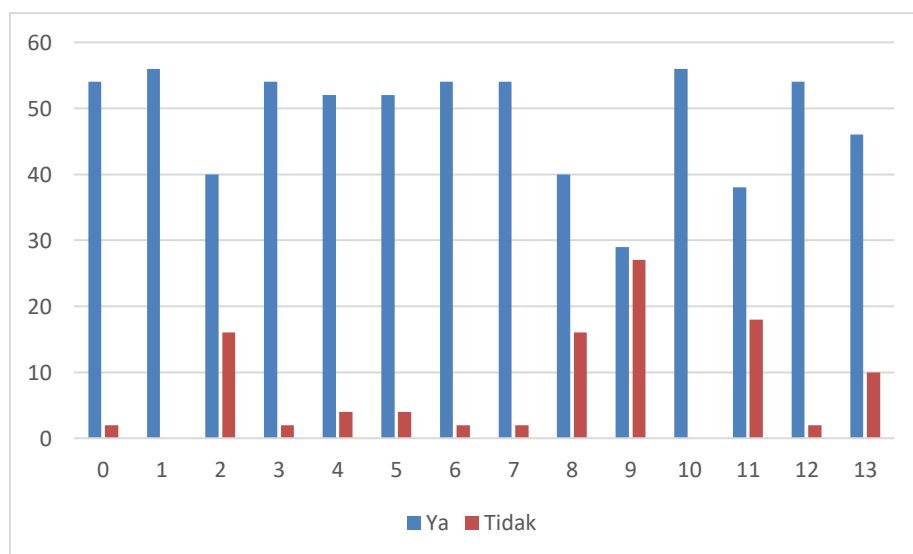
Lampiran 3 adalah hasil estimasi posisi objek yang dilakukan pada gedung bertingkat yaitu gedung D. Estimasi posisi dilakukan dengan mengambil 14 data sampel masing-masing label diambil 1 data sampel dimana untuk mengukur seberapa akuratnya posisi objek yang diambil. Hasil yang di dapat dari beberapa sampel yang dilakukan bahwa posisi objek di dalam ruangan dapat diketahui berdasarkan dari data RSSI yang diambil dengan menggunakan perangkat laptop.

Dari data sampel yang telah diuji terdapat beberapa data error atau data tersebut terbaca di label lain. Seperti misalnya pada label 2, label 8 dan label 11 dimana data uji tersebut diambil disamping dinding yang berbatasan dengan ruangan atau label yang disebelahnya sehingga terdapat data yang terbaca di label disebelahnya seperti gambar dibawah ini.

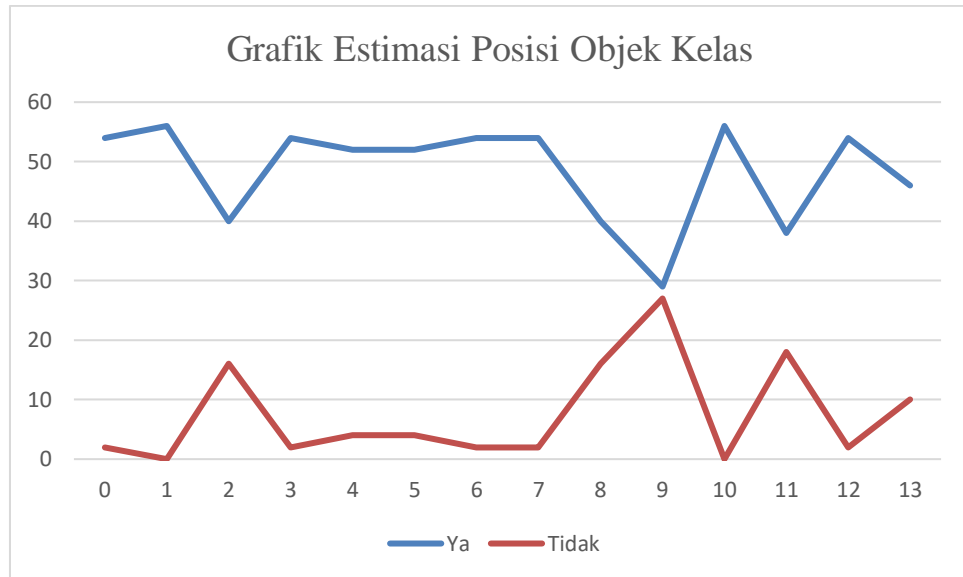
	0	1	2	3	4	5	6
0	0.00801216	0.00062606	0.73513	0.249045	0.000691622	0.00106441	0.000345661
1	0.00801216	0.00062606	0.73513	0.249045	0.000691622	0.00106441	0.000345661
2	0.000148119	0.000262036	0.166761	0.814749	0.00575896	0.000782489	0.00120269
3	0.000148119	0.000262036	0.166761	0.814749	0.00575896	0.000782489	0.00120269
4	0.0225817	0.00169352	0.939407	0.0134557	0.0011448	0.00772876	0.00143231
5	0.0225817	0.00169352	0.939407	0.0134557	0.0011448	0.00772876	0.00143231
6	0.000330606	0.00229094	0.808357	0.188461	0.000266969	2.75534e-05	7.99642e-06
7	0.000330606	0.00229094	0.808357	0.188461	0.000266969	2.75534e-05	7.99642e-06
8	0.00329404	0.000396095	0.885553	0.103882	0.00033602	0.00174232	0.000373799
9	0.00329404	0.000396095	0.885553	0.103882	0.00033602	0.00174232	0.000373799
10	1.62718e-05	5.30377e-05	0.207836	0.789361	0.00027649	9.61522e-05	0.000249283
11	1.62718e-05	5.30377e-05	0.207836	0.789361	0.00027649	9.61522e-05	0.000249283
12	0.00348722	0.000996671	0.959065	0.0301702	0.000293183	0.00114691	0.000824939
13	0.00348722	0.000996671	0.959065	0.0301702	0.000293183	0.00114691	0.000824939
14	0.000758842	0.00514267	0.858198	0.134762	0.000567399	5.41771e-05	1.68797e-05
15	0.000758842	0.00514267	0.858198	0.134762	0.000567399	5.41771e-05	1.68797e-05
16	0.00677509	0.00326016	0.971551	0.0134843	0.000261349	0.00103606	0.000528656
17	0.00677509	0.00326016	0.971551	0.0134843	0.000261349	0.00103606	0.000528656

Gambar 4.7 *Probability* dari prediksi uji sampel

Berdasarkan *probability* seperti gambar diatas kemudian di rangkumlah seluruh 14 data sampel dari hasil *probability* data yang sesuai dengan prediksi dan data yang error. *Probability* yang dimaksud merupakan kemungkinan benar dari suatu data tersebut berada di label sebenarnya.



Gambar 4.8 Hasil Pengujian Estimasi Posisi Objek *Real time*



Gambar 4.9 Hasil Pengujian Estimasi Posisi Objek *Real time*

Pada setiap ruangan terdapat error atau data dengan akurasi rendah atau data RSSI terbaca di label lain. Pada tabel 22 ini merupakan hasil rata-rata *probability* dan error yang didapat dari percobaan sampel yang telah dilakukan.

Tabel 4.18. Rata-rata *Probability* Estimasi Posisi Objek *Real time*

Data	Label	<i>Probability</i>	Missclassification
1	0	95.72%	4.28%
2	1	97.95%	2.05%
3	2	73.71%	26,29%
4	3	94.38%	5,62%
5	4	81.31%	18,69%
6	5	99.94%	0.06%
7	6	97.97%	2,03%
8	7	77.30%	22,70%
9	8	81.81%	18,19%
10	9	77.68%	22,32%
11	10	88.06%	11,94%
12	11	90.49%	9,51%
13	12	80.31%	19,69%
14	13	79.60%	20,40%

Pada tabel diatas terlihat bahwa terdapat beberapa data yang terbaca di label lain sehingga berpengaruh terhadap *probability*. Untuk label 4 dan 9 yang berlokasi di tangga mempunyai *probability* 81.31% dan 77.68%. Untuk di label 2, 7 dan 13 titik pengujiannya berada dinding yang bersebelahan dengan label sebelah sehingga terdapat data yang terbaca di label sebelahnya. Kondisi tersebut mempengaruhi *probability* dari label tersebut. Untuk label yang lain titik pengujian nya berada di tengah ruangan dari lokasi label masing-masing.

4.6 Analisis Perancangan Sistem

Perancangan Sistem estimasi posisi menggunakan *Support Vector Machine* (SVM) merupakan suatu prediksi dari keberadaan suatu objek menggunakan probabilitas. Dimana nilai probabilitas tertinggi merupakan posisi objek. Dalam pelaksanaannya, semakin dekat sinyal *access point* yang diambil untuk data latih maka akan semakin baik akurasi yang akan di dapat. Namun, terdapat juga nilai RSS diluar jangkauan (-100) sehingga mengganggu akurasi data latih. Data latih yang sering berada diluar jangkauan berada di lantai 3 gedung D Fasilkom.

Hasil perancangan system estimasi posisi objek menggunakan metode *fingerprint* menggunakan SVM dengan menggunakan 14 sampel menunjukkan bahwa hasil estimasi posisi dengan *probability* mencapai 93,73% dengan error rata-rata mencapai 6.27%. Analisa yang di dapat dari hasil bisal lebih baik jika *access point* berada ditempat yang bisa di jangkau.