

Deteksi Penyakit Glaukoma pada Citra Fundus Retina Mata Menggunakan Adaptive Thresholding dan Support Vector Machine

AHMAD MUSTOFA

5112100100

Dosen Pembimbing

Prof. Ir. Handayani Tjandrasa, M.Sc., Ph.D.

Bilqis Amaliah, S.Kom., M.Kom.

Pendahuluan

Perancangan & Implementasi

Skenario Uji Coba

Kesimpulan & Saran



Latar Belakang

- Glaukoma merupakan penyebab terbanyak kedua dari kebutaan setelah katarak.
- Kebutuhan akibat glaukoma tidak dapat disembuhkan.
- Glaukoma dapat dikenali dengan menganalisa keberadaan optic disk, optic cup, dan pembuluh darah pada citra retina mata.



Rumusan Masalah

- Bagaimana melakukan segmentasi optic disk, optic cup, dan pembuluh darah dari citra retina mata?
- Bagaimana sistem dapat mendeteksi penderita glaukoma dari citra retina mata?



Batasan Masalah

- Kelas retina mata yang dapat diklasifikasi adalah kelas mata normal dan kelas mata glaukoma.
- Implementasi program dilakukan pada lingkungan komputer desktop dengan menggunakan matlab.



Tujuan

- Merancang dan membuat aplikasi yang bisa melakukan segmentasi optic cup, optic disc, dan pembuluh darah dari citra retina mata.
- Merancang dan membuat aplikasi yang bisa mendeteksi penderita glaukoma dari citra retina mata.

Pendahuluan

Perancangan & Implementasi

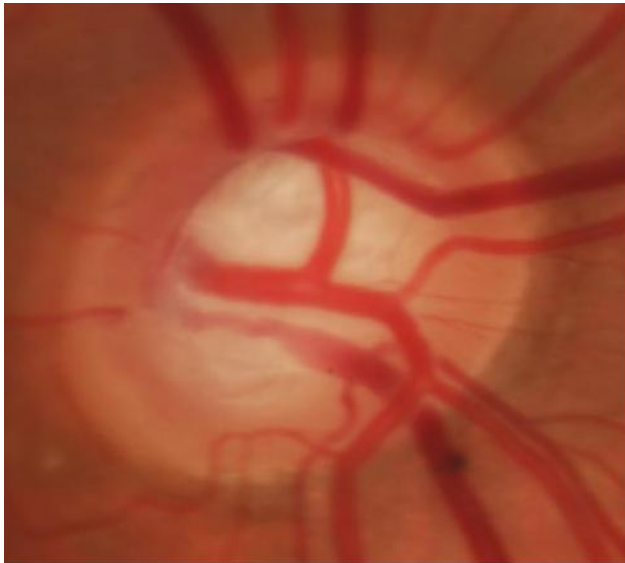
Skenario Uji Coba

Kesimpulan & Saran



Deskripsi Data

- Dataset diambil dari database RIM-ONE.
- Dataset terdiri dari 120 citra retina mata yang terdiri dari 56 mata normal dan 64 mata glaukoma.

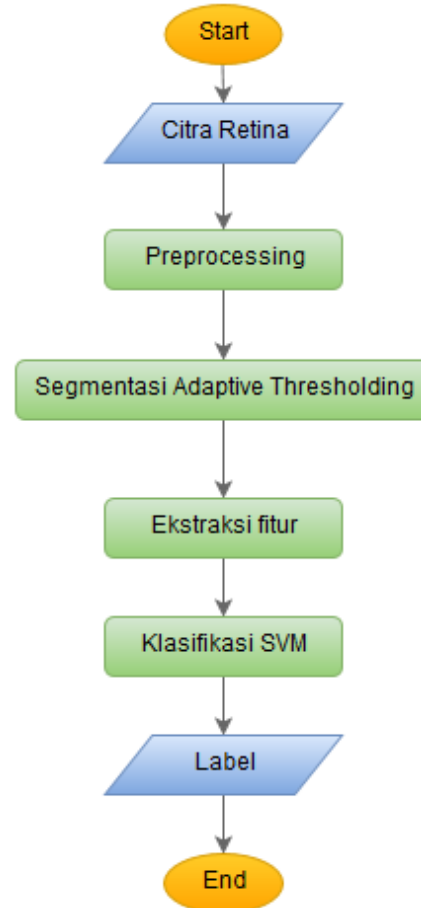


a. Mata glaukoma



b. Mata normal

Diagram Alir Implementasi Secara Umum

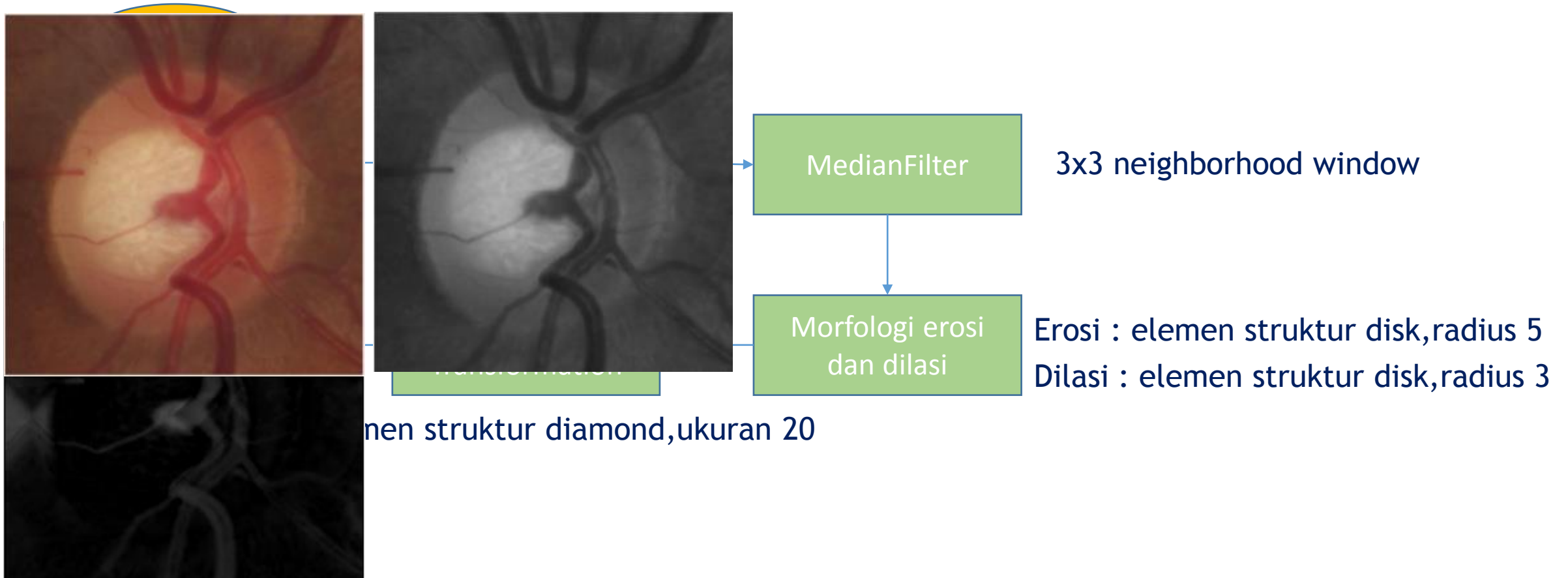




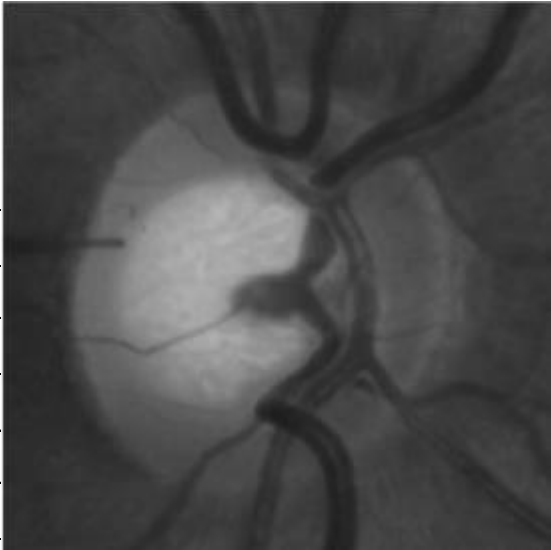
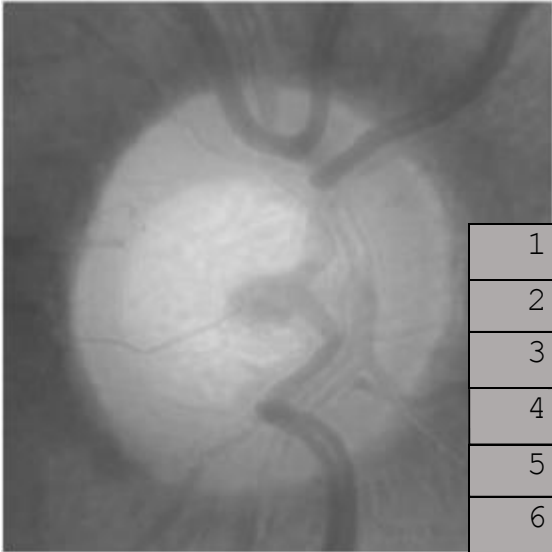
Tahap Preprocessing Data

- Tahap preprocessing terbagi menjadi 2 bagian, yaitu preprocessing pembuluh darah, dan preprocessing optic cup dan optic disk.

Tahap Preprocessing Pembuluh Darah



Tahap Preprocessing Optic Cup dan Optic Disk



Elemen struktur disk,radi

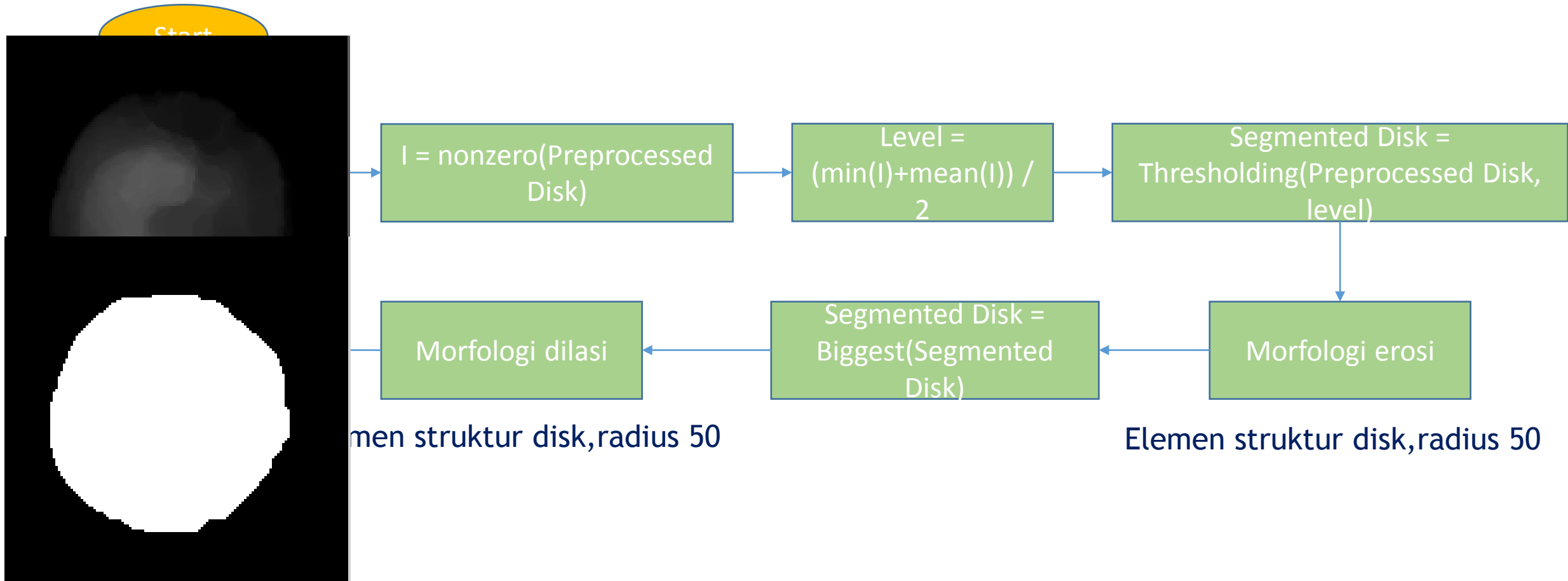
1	Standarize (I, tipe)
2	MEAN = mean (I)
3	Std = StandardDeviation (I)
4	i
5	
6	
7	e
8	
9	
10	e
11	f
12	
13	I = I - minimum (comp)
14	end for
15	return I



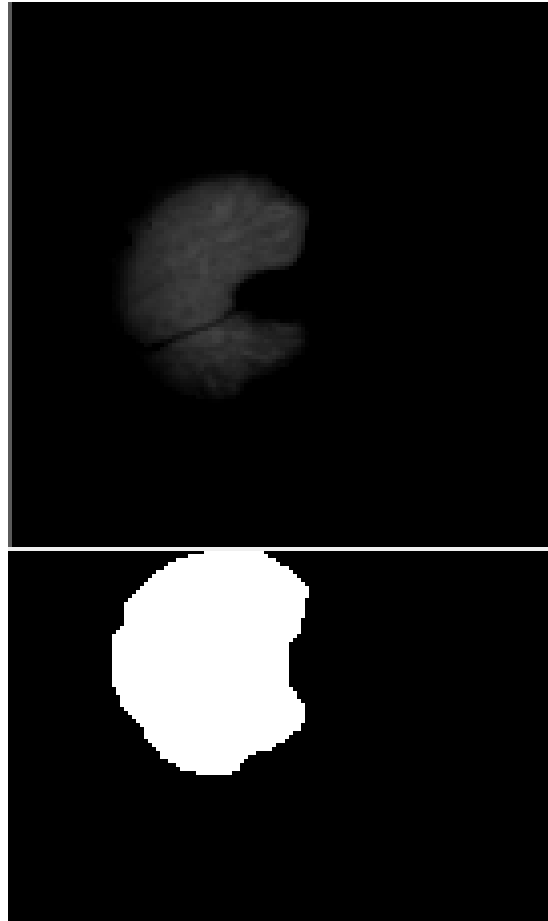
Tahap Segmentasi

- Tahap segmentasi terbagi menjadi 3 bagian, yaitu segmentasi optic disk, segmentasi optic cup, dan segmentasi pembuluh darah.

Tahap Segmentasi Optic Disk



Tahap Segmentasi Optic Cup



Filter Gaussian 1x100, mean 50, standar deviasi 6

Level = otsu(SmoothedHistogram(Preprocessed Cup))

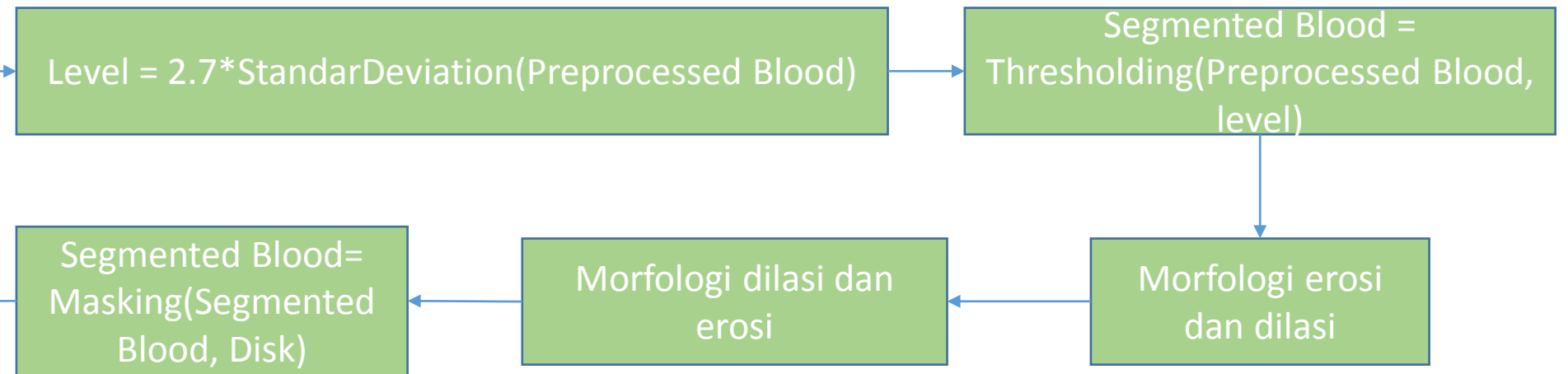
Segmented Cup =
Thresholding(Preprocessed Cup,
level)

Segmented Cup =
Biggest(Segmented Cup)

Morfologi dilasi
dan erosi

Elemen struktur disk, radius 50

Tahap Segmentasi Pembuluh Darah



Elemen struktur diamond, ukuran 5



Tahap Ekstraksi Fitur

1. Cup to Disk Ratio
2. ISNT Pembuluh Darah
3. ISNT Neuro Retinal Rim



Tahap Ekstraksi Fitur CDR (Cup to Disk Ratio)

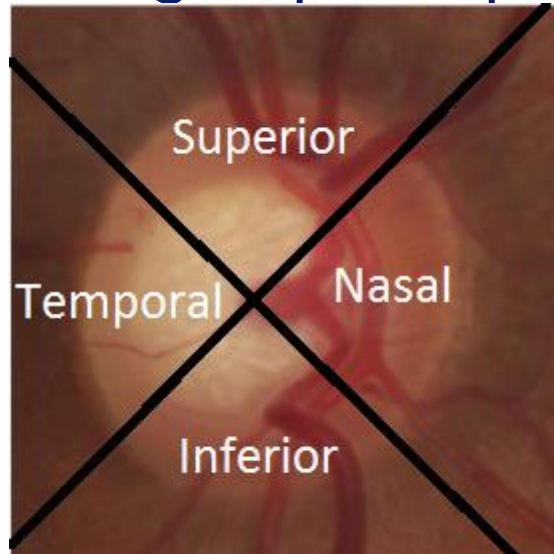
1. Merupakan rasio jejari dari optic cup dan jejari optic disk.
2. Dapat dihitung dengan rumus :

$$CDR = \sqrt{\text{Area of optic cup} / \text{Area of optic disk}}$$



Tahap Ekstraksi Fitur ISNT Pembuluh Darah

- ISNT adalah 4 region pada optic disk, yaitu Inferior, Superior, Nasal, dan Temporal.

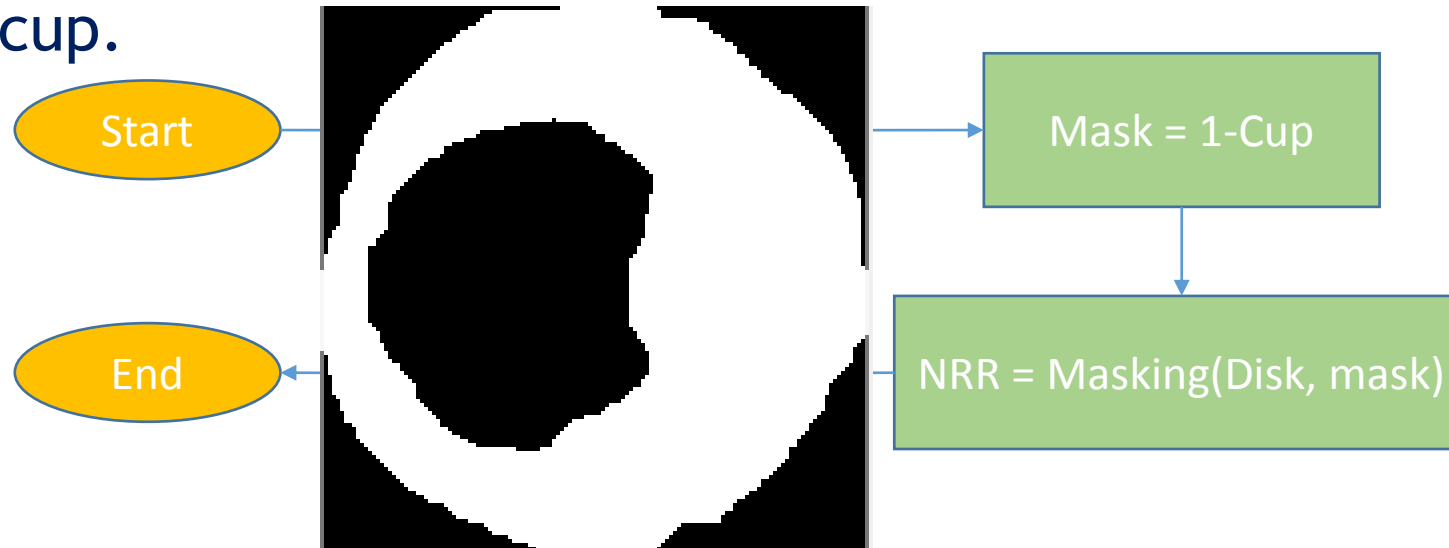


- Fitur ISNT pembuluh darah didapatkan dengan rumus :

$$ISNT = \frac{\text{Luas pembuluh darah pada quadran inferior} + \text{Luas pembuluh darah pada quadran superior}}{\text{Luas pembuluh darah pada quadran nassal} + \text{Luas pembuluh darah pada quadran temporal}}$$

Tahap Ekstraksi Fitur ISNT Neuro Retinal RIM

- Neuro Retinal RIM adalah daerah optic disk yang bukan merupakan optic cup.



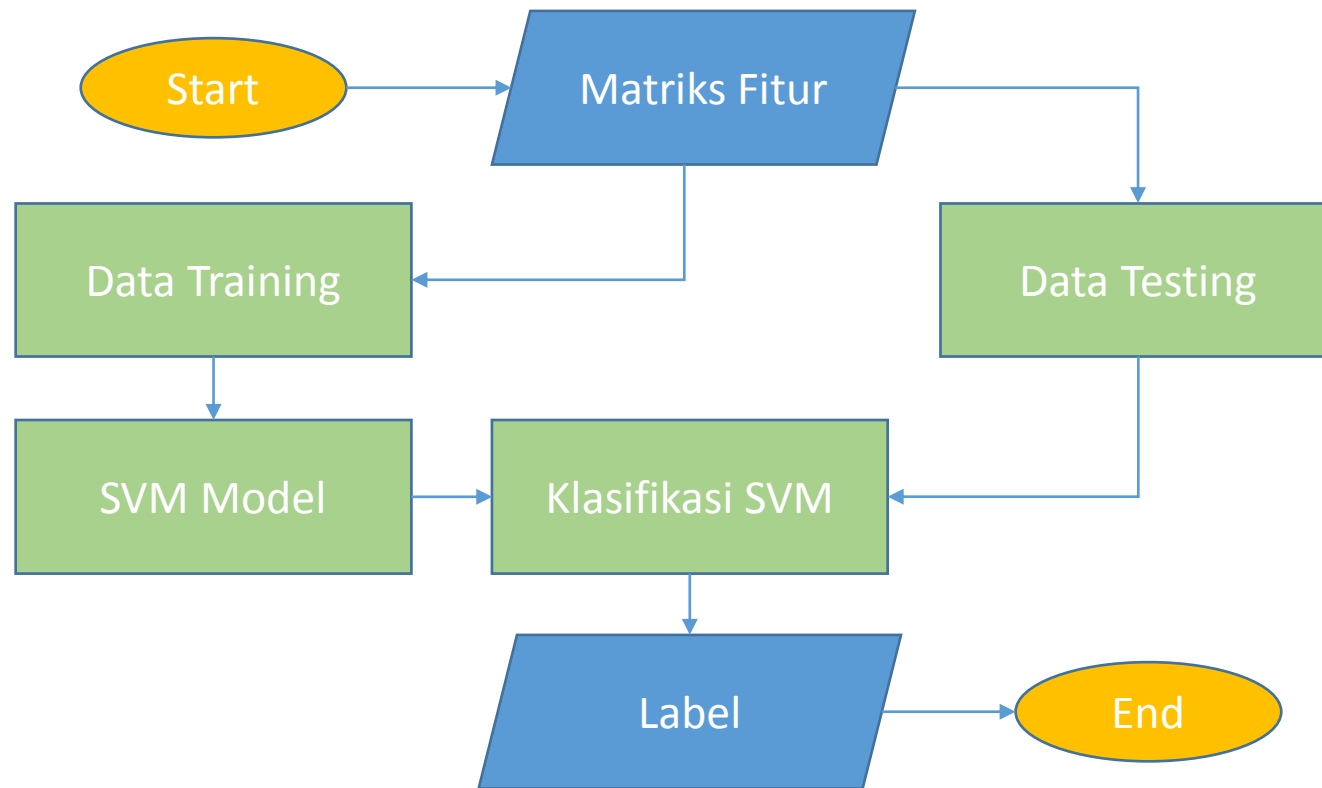
- Fitur ISNT NRR didapatkan dengan rumus :

$$ISNT = \frac{\text{Luas NRR pada quadran inferior} + \text{Luas NRR pada quadran superior}}{\text{Luas NRR pada quadran nassal} + \text{Luas NRR pada quadran temporal}}$$



Tahap Klasifikasi

- Klasifikasi dilakukan dengan menggunakan SVM



Pendahuluan

Perancangan & Implementasi

Skenario Uji Coba

Kesimpulan & Saran



Skenario Uji Coba

- Perhitungan performa meliputi:

$$- akurasi = \frac{TP+TN}{TP+TN+FP+FN}$$

$$- presisi = \frac{TP}{TP+FP}$$

$$- sensitivitas = \frac{TP}{TP+FN}$$

		Prediksi	
		1	0
Aktual	1	TP	FN
	0	FP	TN

- Hasil perhitungan merupakan nilai rata-rata performa dari 5 kali uji coba dengan menggunakan 10-folds cross validation.



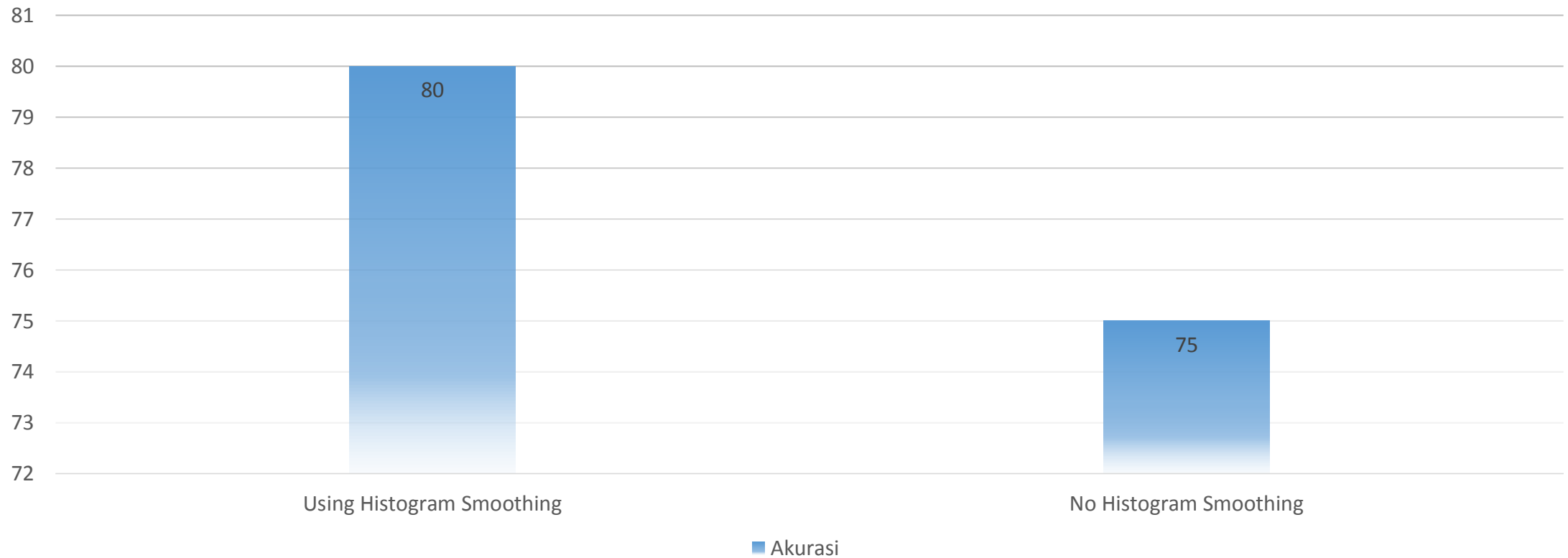
Skenario Uji Coba 1

- Segmentasi optic cup dengan dan tanpa Histogram Smoothing
- Fungsi kernel *SVM* = *Linear*, *SVM method* = *SMO*



Skenario Uji Coba 1

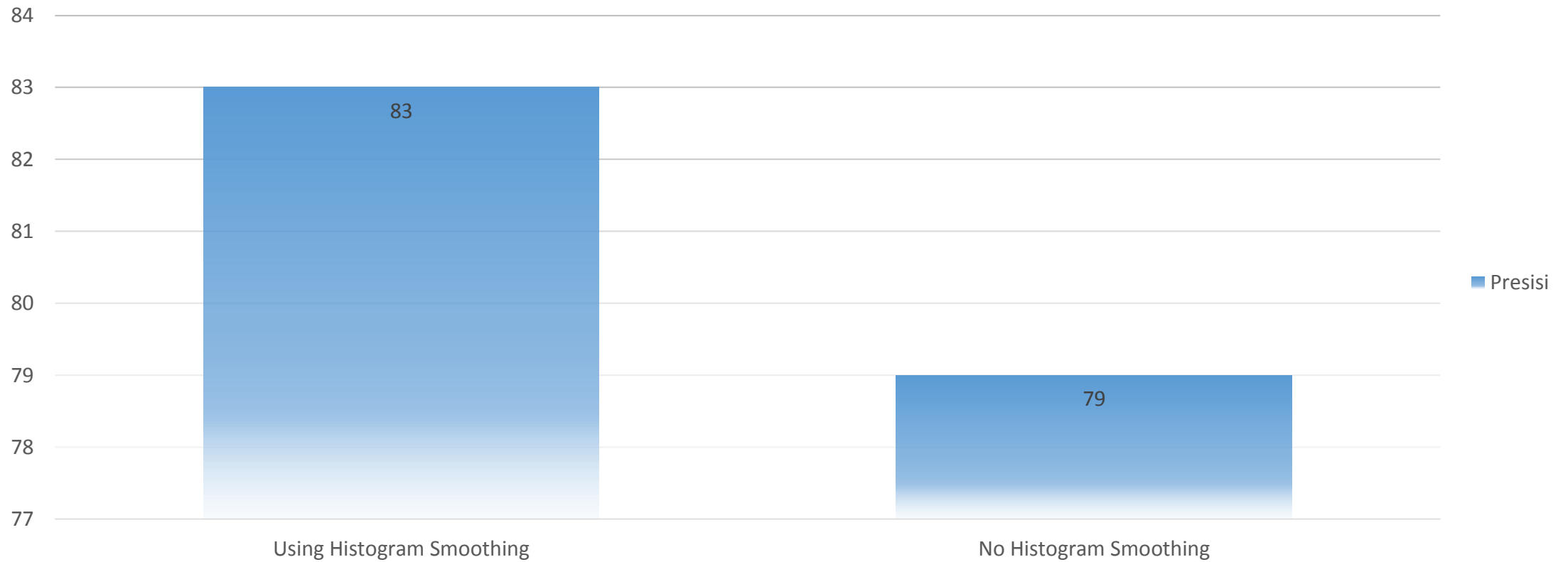
AKURASI





Skenario Uji Coba 1

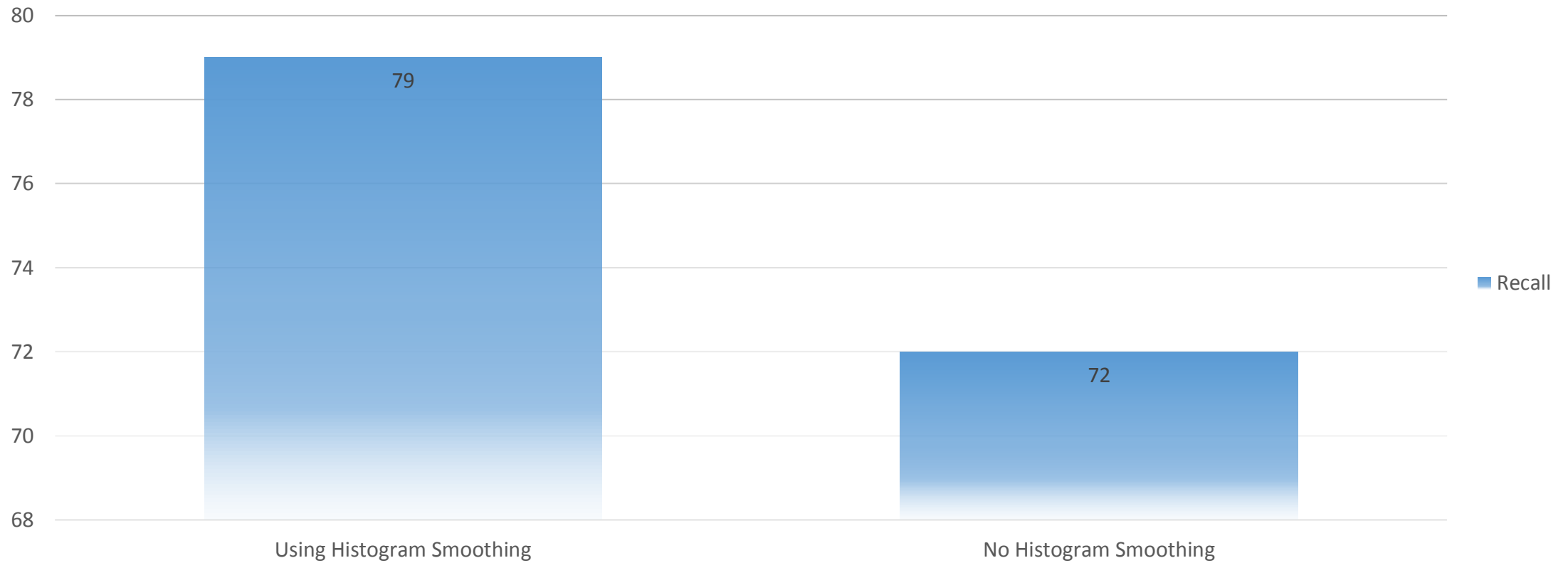
PRESISI





Skenario Uji Coba 1

RECALL





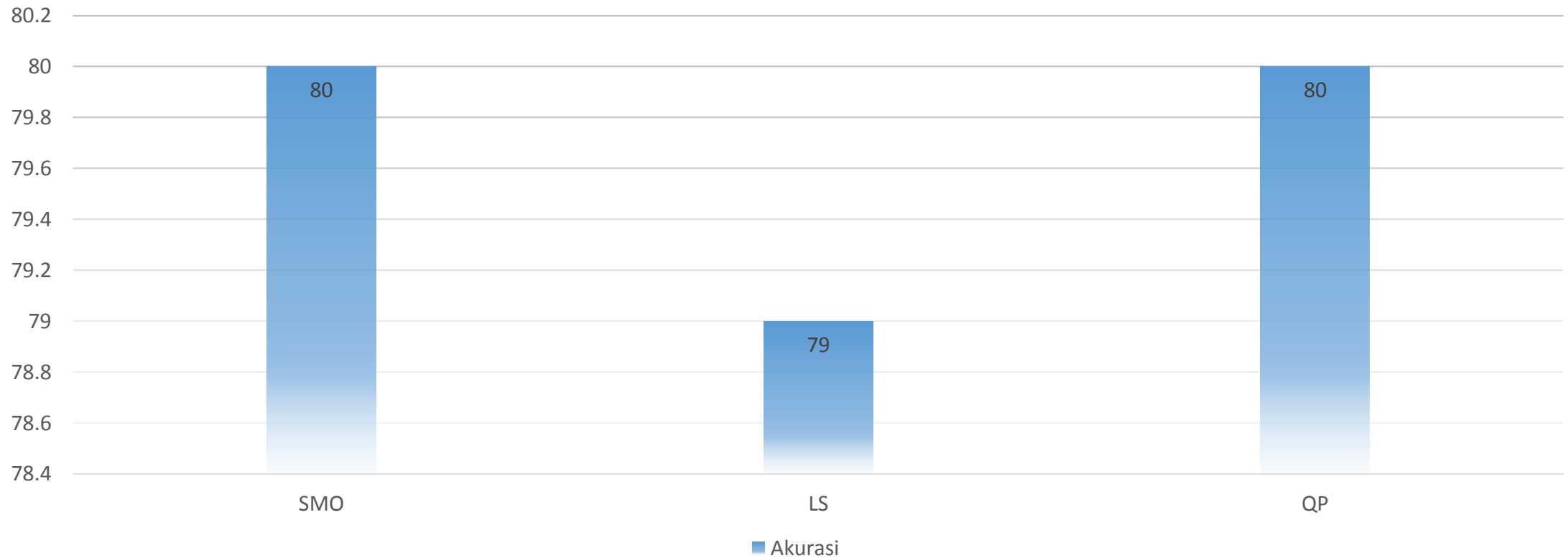
Skenario Uji Coba 2

- Variasi SVM *method*
 - a. SMO (Sequential Minimum Optimization)*
 - b. LS (Least Squares)*
 - c. QP (Quadratic Programming)*



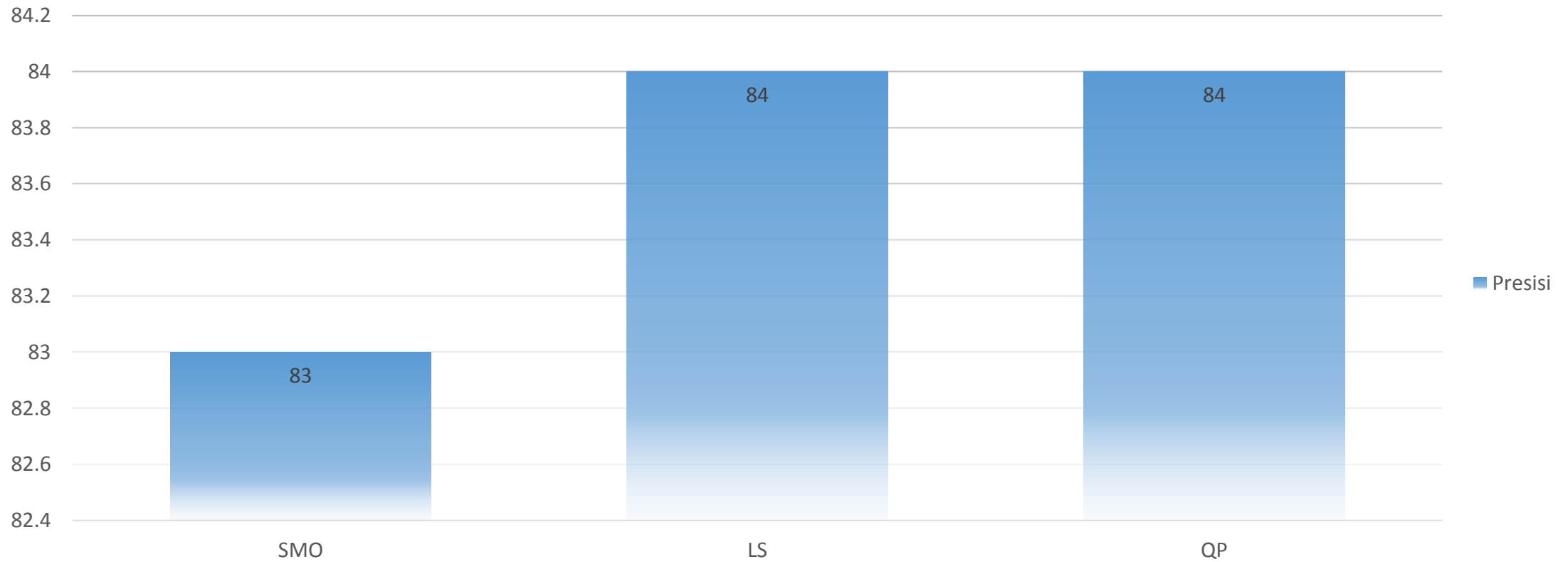
Skenario Uji Coba 2

AKURASI



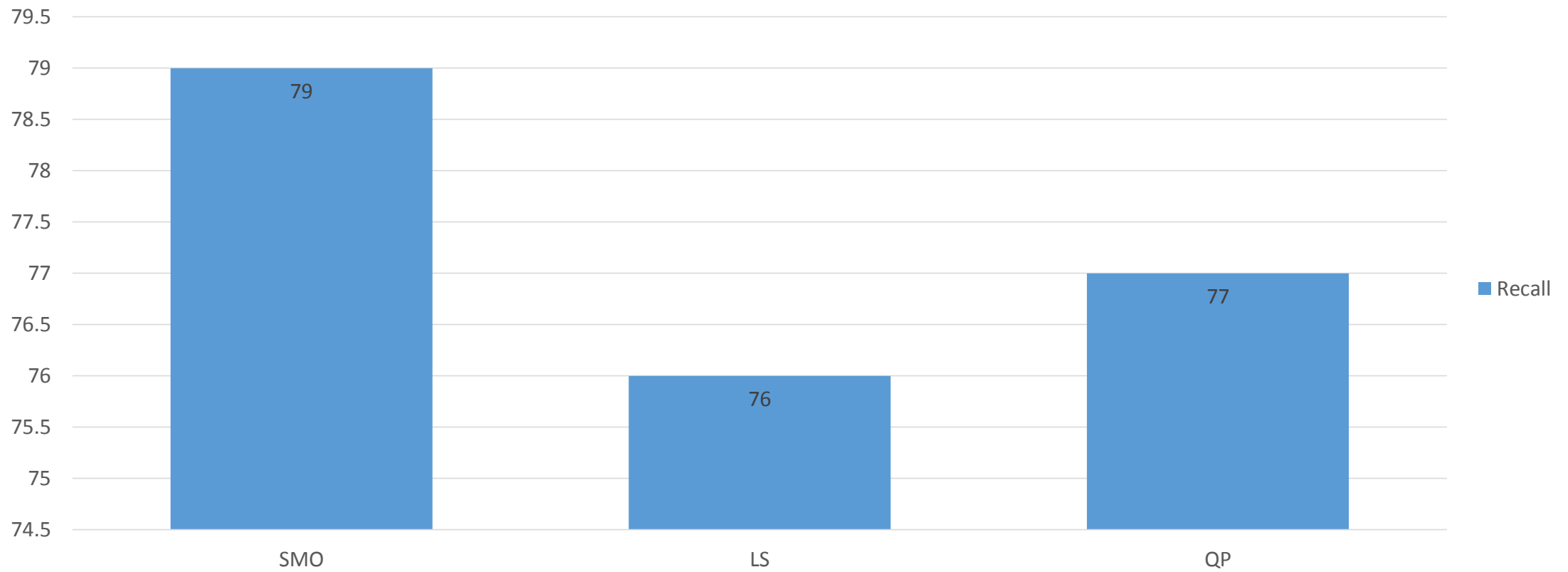
Skenario Uji Coba 2

PRESISI



Skenario Uji Coba 2

Recall



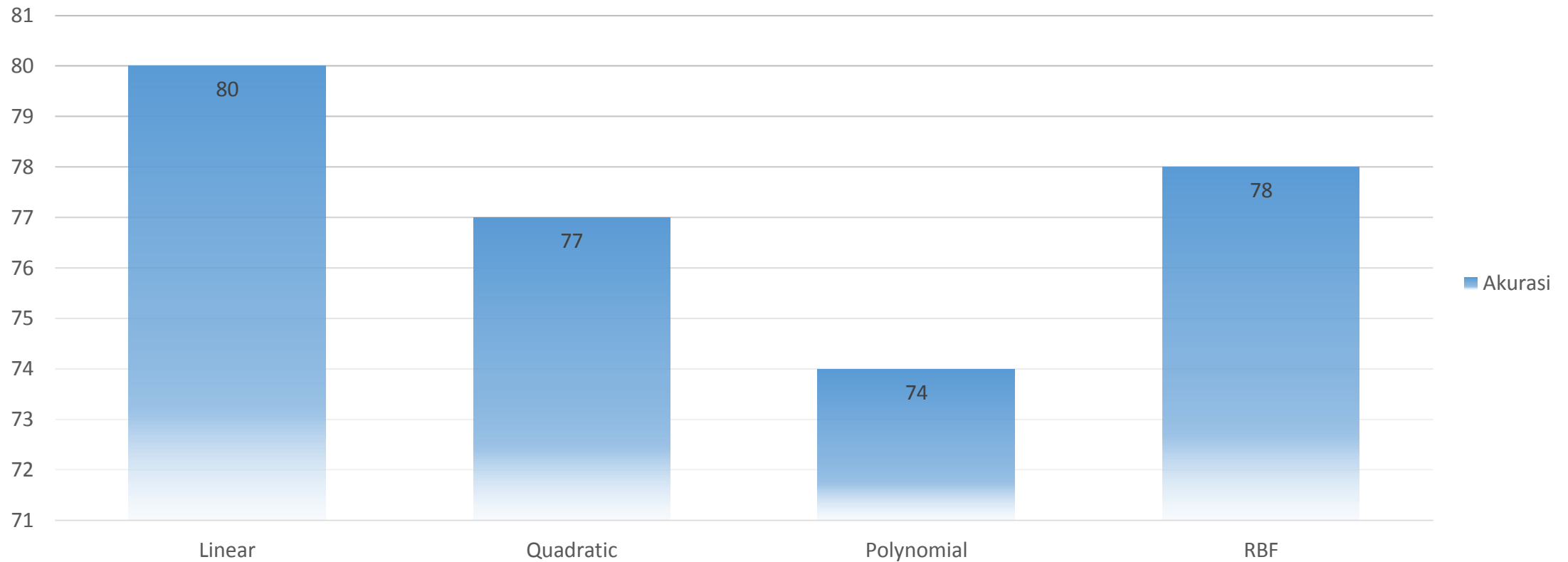


Skenario Uji Coba 3

- Variasi fungsi kernel
 - a. Linear*
 - b. Quadratic*
 - c. Polynomial orde 3*
 - d. RBF (Gaussian Radial Basis Function)*

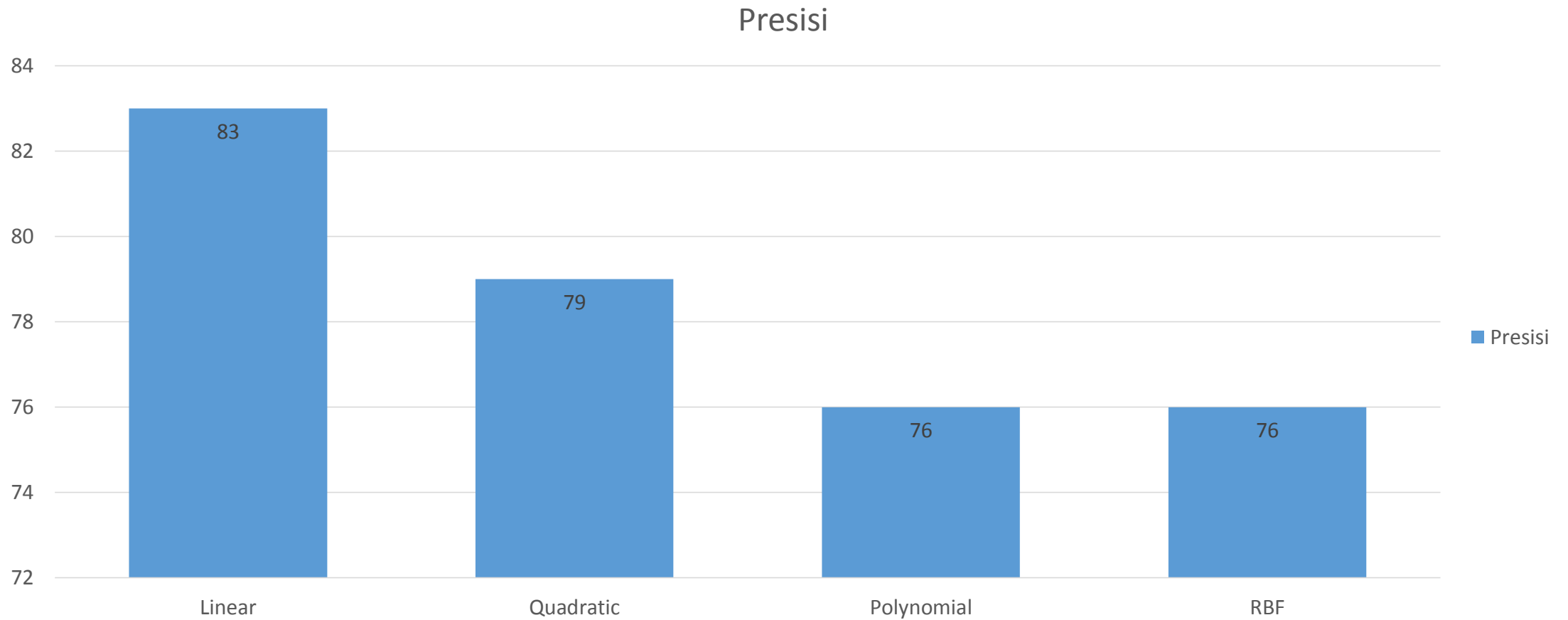
Skenario Uji Coba 3

AKURASI

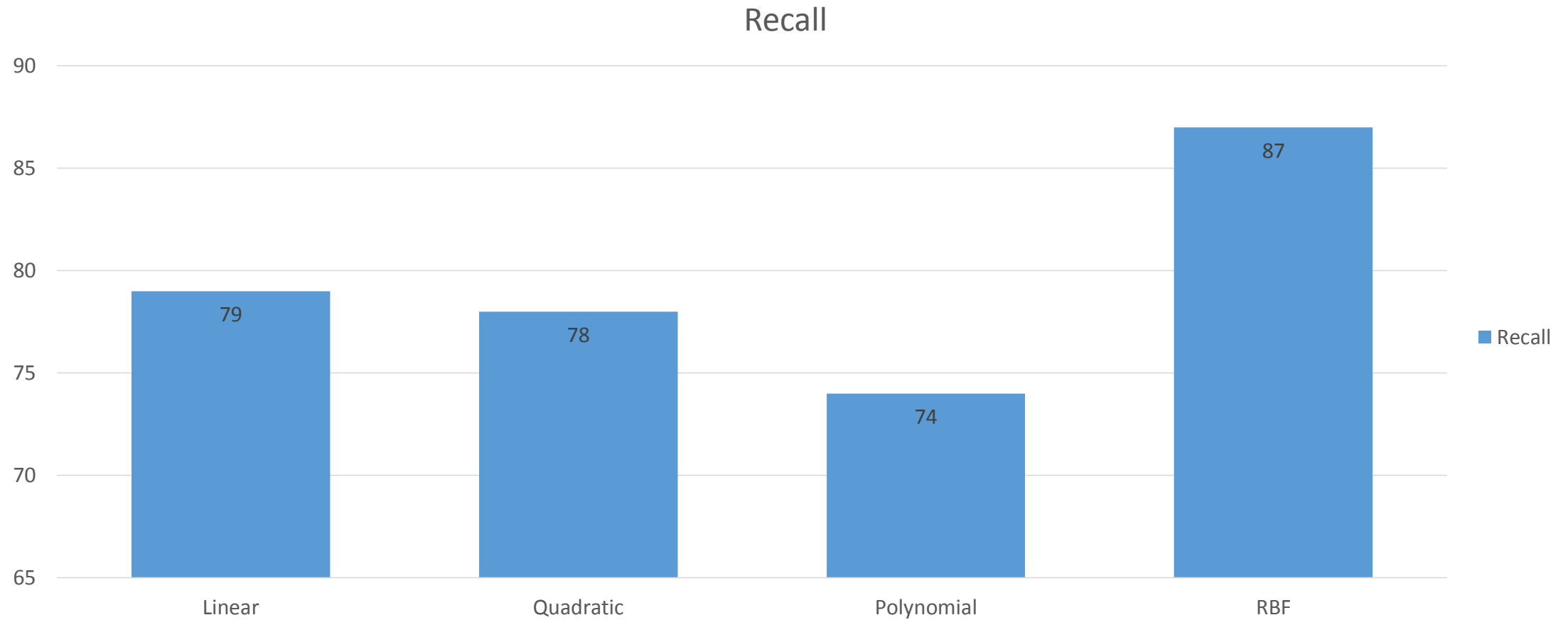




Skenario Uji Coba 3



Skenario Uji Coba 3



Pendahuluan

Perancangan & Implementasi

Skenario Uji Coba

Kesimpulan & Saran



Kesimpulan

- Keberadaan pembuluh darah dan pantulan cahaya dapat mengganggu proses segmentasi optic disk dan optic cup.
- Proses smoothing histogram dapat meningkatkan performa sistem sebesar 5%.
- Metode pencarian hyperplane yang menghasilkan performa terbaik pada support vector machine adalah sequential minimal optimization.
- Fungsi kernel yang menghasilkan performa terbaik pada support vector machine adalah fungsi kernel linear.



Saran

- Perlu dikembangkan metode untuk menangani noise pantulan cahaya pada citra retina mata.
- Perlu dikembangkan metode penghilangan pembuluh darah yang lebih baik pada segmentasi optic cup.
- Diperlukan penelitian lebih lanjut untuk mendapatkan fitur lain yang lebih baik daripada fitur yang digunakan dalam tugas akhir ini.

Terima Kasih



K-Fold Cross Validation

- Dataset dibagi secara acak menjadi K bagian berukuran sama
- Satu bagian digunakan sebagai data uji, dan K-1 bagian lainnya digunakan sebagai data latih.
- Proses tersebut diulang sebanyak K kali.
- Masing-masing bagian digunakan tepat satu kali sebagai data uji.





Median Filter

- Pseudocode:

1	MedianFiltering(I, size)
2	foreach p = pixels in I
3	p = median(part(I(position(p).x - size, position(p).y + size))
4	end foreach
5	return I



Bottom Hat Transformation

- Rumus :

$$B_{hat}(I) = ((I \oplus b) \ominus b) - I$$

dimana \oplus adalah operasi dilasi dan \ominus adalah operasi erosi.



Otsu Thresholding

- Metode segmentasi berbasis clustering sehingga didapatkan batas yang dapat dijadikan acuan untuk melakukan segmentasi pada suatu citra.
- Jumlah kelas yang dapat ditangani hanya 2, yaitu objek dan *background*.

$$\sigma_b^2(t) = \omega_0(t)\omega_1(t)[\mu_0(t) - \mu_1(t)]^2 \text{ dimana } \omega_0(t) = \sum_0^{t-1} p(i) \text{ dan } \omega_1(t) = \sum_t^{255} p(i) \\ \text{serta } \mu_0(t) = \frac{\sum_0^{t-1} ip(i)}{\omega_0} \text{ dan } \mu_1(t) = \frac{\sum_t^{255} ip(i)}{\omega_1} \text{ dengan } p(i) = \frac{\text{histogram}(i)}{\text{jumlah pixel}}$$

- Pseudocode:

Hitung histogram dan peluang dari setiap level intensitas

Inisialisasi nilai awal $\omega_i(0)$ dan $\mu_i(0)$

For t = 1 to 256

 Update ω_i dan μ_i

 Hitung $\sigma_b^2(t)$

Pilih t dengan nilai $\sigma_b^2(t)$ maksimum



Support Vector Machine (SVM)

- Mencari *hyperplane* yang optimal dilakukan dengan persamaan

$$\frac{1}{2} ||w||^2 + C \sum_{i=1}^n \xi_i$$

- Persamaan di atas dapat dipecahkan dengan teknik komputasi *Lagrange Multiplier*

$$L(w, b, \alpha) = \frac{1}{2} ||w||^2 - \sum_{i=1}^n \alpha_i (y_i ((x_i \cdot w + b) - 1))$$

- Persamaan *Lagrange* dapat dimodifikasi sebagai maksimalisasi yang hanya mengandung α



Support Vector Machine (SVM)

- Persamaan *Lagrange* dapat dimodifikasi sebagai maksimalisasi yang hanya mengandung α

$$\sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j x_i x_j$$
$$\alpha_i \geq 0 \ (i = 1, 2, \dots, n) \quad \sum_{i=1}^n \alpha_i y_i = 0$$



Suport Vector Machine (SVM)

- SVM pada dasarnya adalah sebuah model klasifikasi linear, namun bisa digunakan untuk melakukan klasifikasi non-linear menggunakan *kernel trick*.
- *Kernel trick* dilakukan dengan cara melakukan transformasi ruang data asli menjadi ruang data dengan dimensi lebih tinggi.

Jenis kernel	Definisi
Polynomial	$K(x_i, x_j) = (x_i \cdot x_j + 1)^p$
Gaussian	$K(x_i, x_j) = \exp\left(\frac{\ x_i - x_j\ ^2}{2\gamma^2}\right)$

- Persamaan *hyperplane*

$$f(\Phi(x)) = \sum_{i=1, x_i \in SV}^n \alpha_i y_i K(x, x_i) + b$$



Least Squares

- Salah satu variasi metode untuk menyelesaikan komputasi Quadratic Programming pada SVM

$$L(w, b, e, \alpha) = \frac{1}{2} ||w||^2 - \sum_{i=1}^n \alpha_i (y_i ((x_i \cdot w + b) - 1 + e_i))$$

dimana e_i adalah nilai *least squares* dari y_i, x_i .



- Salah satu variasi metode untuk menyelesaikan komputasi Quadratic Programming pada SVM
- Dilakukan dengan memecah komputasi *Quadratic Programming* menjadi beberapa sub kecil. Sehingga kompleksitas komputasi *Quadratic Programming* pada SVM dapat direduksi.