



# Identifikasi Citra berdasarkan Gigitan Ular menggunakan Metode *Active Contour Model* dan *Support Vector Machine*

Dhiya Ulhaq Dewangga, Adiwijaya, Dody Qori Utama

Fakultas Teknik Informatika, Informatika, Universitas Telkom, Bandung, Indonesia  
Email: [dhiyaud@students.telkomuniversity.ac.id](mailto:dhiyaud@students.telkomuniversity.ac.id), [adiwijaya@telkomuniversity.ac.id](mailto:adiwijaya@telkomuniversity.ac.id),  
[dodyqori@telkomuniversity.ac.id](mailto:dodyqori@telkomuniversity.ac.id)

## Abstrak

Negara tropis yang memiliki iklim hangat dan lembab merupakan habitat yang cocok bagi kehidupan hewan reptil, terutama ular. Ular merupakan jenis hewan melata yang banyak terdapat di negara tropis terutama di Indonesia. Hal terburuk yang terjadi jika bertemu ular adalah terkena gigitannya. Jika gigitan berasal dari ular yang berbisa dapat mengakibatkan masalah yang lebih serius daripada gigitan oleh ular tidak berbisa, dimana dapat mengakibatkan kelumpuhan, cacat, dan yang terburuk adalah kematian. Menurut WHO (World Health Organization) diperkirakan 5,4 juta orang digigit ular setiap tahunnya dengan hampir 2,7 juta terkena gigitan ular berbisa dan terkena gejalanya. Sekitar 81.000 hingga 138.000 orang meninggal setiap tahun. Penelitian ini menggunakan sarana teknologi berbasis image processing untuk membuat sistem identifikasi gigitan ular apakah berbisa atau tidak berbisa. Metode yang digunakan pada sistem ini yaitu *Active Contour Model* dan *Support Vector Machine*. Dengan menggunakan metode-metode tersebut, didapat akurasi tertinggi pada kernel SVM terbaik yaitu pada kernel RBF dan kernel Polynomial.

**Kata Kunci:** Ular, Gigitan, Pengolahan Citra, *Active Contour Model*, *Support Vector Machine*

## Abstract

Tropical countries have a warm and humid climate are suitable habitat for the lives of reptile animals, especially snakes. Snakes are a type of reptile animal that is widely found in tropical countries, especially in Indonesia. The worst thing that happens when meeting a snake is the bite of snake. If the bite comes from a venomous snake it can cause a more serious problem than the bite from non-venomous snake is, which can cause paralysis, disability, and the worst is death. According to the WHO (World Health Organization) an estimated 5.4 million people are bitten by snakes each year with almost 2.7 million being bitten by venomous snakes and get affected symptoms. Around 81,000 to 138,000 people die every year. This research uses image processing technic to make the identification system of snake bites whether venomous or non-venomous. The method used in this system is *Active Contour Model* and *Support Vector Machine*. By using these methods, the highest accuracy is obtained in the best of SVM kernel, on RBF kernel and Polynomial kernel.

**Keywords:** Snake, Bite, Image Processing, *Active Contour Model*, *Support Vector Machine*

## 1. PENDAHULUAN

Negara tropis yang memiliki iklim hangat dan lembab merupakan habitat yang cocok bagi kehidupan hewan reptil, terutama ular. Ular merupakan jenis hewan melata yang banyak terdapat di negara tropis terutama di Indonesia. Ular dapat dibedakan menjadi dua yaitu ular berbisa dan ular tidak berbisa. Ciri utama yang dapat dilihat dari ular berbisa adalah memiliki sepasang taring pada bagian rahang atas [1].

Hal terburuk yang terjadi jika bertemu ular adalah terkena gigitannya. Gigitan ular merupakan masalah serius dan keadaan darurat yang sering terabaikan dimana dapat mengakibatkan luka ringan hingga mengancam nyawa. Jika gigitan berasal dari ular yang berbisa dapat mengakibatkan masalah yang lebih serius daripada gigitan oleh ular tidak berbisa, dimana dapat mengakibatkan kelumpuhan yang menghambat pernapasan, pendarahan fatal, gagal ginjal, kerusakan jaringan yang mengakibatkan cacat permanen hingga amputasi anggota badan, dan yang terburuk adalah kematian. Gigitan ular lebih banyak terjadi di negara tropis dan subtropis, dimana umumnya yang terkena gigitan adalah petani dan anak-anak.

Menurut WHO (*World Health Organization*) diperkirakan 5,4 juta orang digigit ular setiap tahunnya dengan hampir 2,7 juta terkena gigitan ular berbisa dan terkena gejalanya. Sekitar 81.000 hingga 138.000 orang meninggal setiap tahun akibat gigitan ular, dan sekitar tiga kali lebih banyak terkena amputasi dan cacat permanen yang disebabkan oleh gigitan ular setiap tahunnya [2].

Meski tidak ada data resmi gigitan ular di Indonesia, bisa diperkirakan korban akibat gigitan ular mencapai 135.000. Jumlah itu baru yang dilaporkan ke rumah sakit dan Puskesmas (Pusat Kesehatan Masyarakat), sedangkan yang tidak dilaporkan bisa lebih banyak [3].

Berdasarkan latar belakang tersebut, penulis membuat sistem mengenai gigitan ular yang dapat memprediksi apakah gigitan ular tersebut berbahaya atau tidak. Prediksi dilakukan menggunakan citra dan diolah agar citra tersebut dapat memberikan hasil prediksi. Dengan adanya sistem ini diharapkan dapat membantu menggolongkan gigitan ular yang termasuk ke dalam gigitan yang berbahaya atau tidak sehingga mendapat tindakan yang seharusnya diambil agar meminimalisir efek terburuk yaitu kematian.

Segmentasi yang dilakukan pada citra yaitu dengan menggunakan *Active Contour Model*. Penggunaan metode berdasarkan kelebihanannya yaitu dapat menyesuaikan pola objek sesuai dengan parameter masukannya sehingga akan mengecilkan daerah yang akan dianalisis. Hal ini akan meningkatkan akurasi serta mempercepat waktu komputasi pada citra [4].



Salah satu metode yang cukup terkenal untuk melakukan prediksi dan digunakan pada penelitian ini adalah *Support Vector Machine* (SVM). Kelebihan dari SVM diantaranya adalah dalam menentukan jarak yang menggunakan support vector, sehingga proses komputasi menjadi lebih cepat [5]. Selain itu kelebihan SVM lainnya terdapat pada penelitian dilakukan oleh Aris [6] yaitu membandingkan metode *K-Nearest Neighbor* (KNN) dengan metode *Support Vector Machine* (SVM) untuk melakukan klasifikasi, dan diperoleh kesimpulan bahwa SVM memiliki kinerja yang lebih unggul karena mampu mengklasifikasikan data dalam pengenalan plat kendaraan dengan ketepatan sebesar 95%.

## 2. METODE PENELITIAN

### 2.1 Jenis dan Sumber Data

Data yang digunakan berjumlah 20 data yang terdiri dari citra gigitan ular tidak berbisa sejumlah 7 data, dan citra gigitan ular berbisa sejumlah 13 data. Setiap data berukuran  $400 \times 400$  pixels. Data dibagi menjadi data *train* dan data *test* dengan rincian 14 data *train* yang terdiri dari 6 citra ular tidak berbisa dan 8 citra gigitan ular berbisa, serta 6 data *test* yang terdiri dari 2 citra gigitan ular tidak berbisa dan 4 data ular berbisa. Data didapat dari penelitian sebelumnya yang dilakukan oleh Rayiemas [7].

### 2.2 Active Contour Model untuk preprocessing

*Active Contour Model* atau dapat juga disebut *Snake* model merupakan suatu model yang digunakan untuk melakukan segmentasi objek pada citra. Yang membedakan model *Active Contour* ini dengan model segmentasi lainnya adalah proses minimizing kurva pada proses segmentasi [8]. Pada prosesnya, *Active Contour Model* akan membuat suatu *initial contour* yang mengelilingi objek, kemudian dengan adanya nilai dari suatu objek citra akan menyebabkan kurva mengecil dan mengikuti pola dari objek tersebut. Kurva dapat bergerak mendekati ke arah objek dan menyesuaikan dengan bentuk objek karena adanya nilai pada kurva. Nilai yang terdapat pada kurva bertanggung jawab untuk mengubah bentuk kurva sesuai dengan bentuk objek yang diinginkan.

Metode *Active Contour Model* dilakukan pada tahap *preprocessing*, dimana pada proses ini data asli akan dipersiapkan untuk diolah dan diekstraksi fiturnya. Dalam hal ini, *preprocessing* memiliki tujuan antara lain: mengubah citra RGB menjadi *grayscale*, mengubah citra menjadi biner, menghapus *noise* atau objek yang tidak diperlukan pada citra, dan melakukan segmentasi dengan menggunakan *Active Contour Model*. Terdapat proses lanjutan dari *preprocessing* yang digunakan agar hasil pada proses *preprocessing* yang kurang dapat dimaksimalkan untuk digunakan pada tahap berikutnya yaitu pengekstrasian fitur pada citra.

Fitur atau ciri merupakan karakteristik unik dari suatu objek. Sedangkan, proses ekstraksi fitur atau ekstraksi ciri adalah proses untuk mendapatkan ciri pembeda yang membedakan suatu objek dari objek yang lain (*discrimination*) [9]. Ekstraksi fitur dilakukan dengan menyeleksi fitur, dimana merupakan teknik yang digunakan untuk memilih fitur terbaik dari sekian banyak fitur yang tersedia pada sebuah data [10]. Ekstraksi ciri terbagi menjadi tiga macam yaitu : Ekstraksi ciri bentuk dan kontur dimana fitur dikategorikan berdasarkan batas (*boundary-based*) dan berdasarkan daerah (*region-based*), lalu ada ekstraksi ciri tekstur dimana fitur pembeda adalah tekstur yang merupakan karakteristik penentu pada citra, dan yang terakhir adalah ekstraksi ciri warna dimana ciri pembeda adalah warna yang memiliki komposisi warna RGB (*Red Green Blue*) [11]. Pada penelitian ini fitur yang diambil adalah rata-rata jarak antar gigitan dan jumlah titik gigitan.

### 2.3 Support Vector Machine untuk Klasifikasi

Klasifikasi yang digunakan pada penelitian ini adalah *Support Vector Machine* (SVM). Dikembangkan oleh Boser, Guyon, dan Vapnik. dan diperkenalkan pada tahun 1992 di *Annual Workshop on Computational Learning Theory*, konsep dasar dari SVM sebenarnya merupakan gabungan atau kombinasi dari teori-teori komputasi yang telah ada pada tahun sebelumnya, seperti *margin hyperplane* [12]. SVM adalah suatu klasifikasi yang pada dasar perhitungannya diambil dari pembelajaran statistik dan hasil dari klasifikasi ini lebih baik jika dibandingkan dengan klasifikasi yang lain. Ciri SVM adalah *low generation error*, secara komputasi tidak mahal, mudah untuk diinterpretasikan, namun sensitif pada *tunning parameter* dan pilihan kernel [13]. SVM merupakan metode yang sangat baik digunakan untuk klasifikasi dan regresi. SVM merupakan algoritma *supervised* yang merupakan algoritma pembelajaran secara terarah dengan melihat kelas atau label dari data [14].

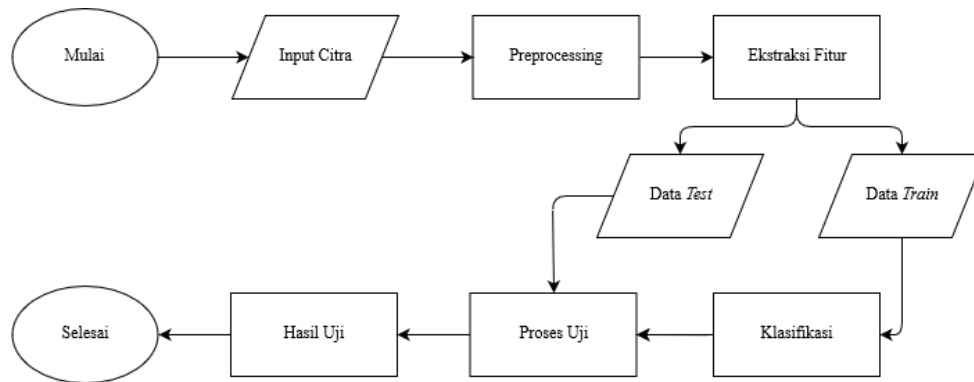
Ide dasar dari SVM adalah memaksimalkan batas *hyperplane*. *Hyperplane* adalah garis pemisah terbaik antara kelas. *Hyperplane* dapat ditemukan dengan cara menghitung margin *hyperplane* dan mencari titik maksimalnya. Margin adalah jarak antara *hyperplane* dengan data terdekat dari masing-masing kelas. Data yang paling dekat biasa disebut dengan *Support Vector* [15].

Terdapat teknik pengklasifikasian pada SVM yang disebut fungsi kernel. Teknik fungsi kernel membantu mentransformasikan bidang masukkan yang sebelumnya berdimensi rendah ke dimensi yang lebih tinggi. Menurut Hsu, beberapa fungsi Kernel yang umumnya digunakan adalah *Radial Basis Function* (RBF), *Polynomial*, dan *Linear*. Kernel RBF adalah kernel yang umum digunakan untuk data yang sudah valid (*available*) dan merupakan default dalam tools SVM. Kernel *Polynomial* adalah kernel yang sering digunakan untuk klasifikasi citra. Kernel *Linear* adalah kernel yang paling sederhana dari semua fungsi kernel [16].



## 2.4 Perancangan Sistem

Penelitian ini membangun sistem untuk mengenali gigitan ular pada citra dan selanjutnya citra tersebut dikategorikan gigitan ular berbisa atau tidak berbisa. *Input* dari sistem berupa sebuah citra yang menggambarkan pola gigitan yang terdapat pada suatu media, dan hasil *output* berupa kategori hasil identifikasi bekas gigitan yang didapat. Secara garis besar, sistem yang akan digunakan terdiri dari empat tahap, yaitu: tahap *preprocessing*, tahap ekstraksi fitur pada citra, tahap pelatihan klasifikasi dan terakhir tahap pengujian dimana data akan diidentifikasi apakah termasuk gigitan ular berbisa atau tidak. Alur pada sistem ini ditunjukkan pada Gambar 1.



**Gambar 1.** Alur Sistem

Diawal tahap, semua citra yang digunakan akan memasuki tahap *preprocessing* terlebih dahulu sebelum digunakan untuk mengekstraksi fitur dari citra. Masukan dari tahap ini merupakan citra RGB gigitan ular, dan keluaran atau hasil dari proses ini adalah kumpulan citra luka gigitan yang berupa citra biner atau hitam dan putih. Pada tahap ini proses pertama yang dilakukan adalah membersihkan citra dari berbagai gangguan atau *noise*, lalu mengubah citra menjadi hitam dan putih, dan selanjutnya digunakan metode *Active Contour Model* untuk melakukan *filtering* pada citra dan membantu mengenali objek titik-titik gigitan. Setelah melakukan *Active Contour Model* selanjutnya dilakukan proses lanjutan pada citra gigitan ular berbisa agar didapat objek titik gigitan terbesar sebanyak 2 titik gigitan, sedangkan pada citra gigitan ular tidak berbisa tidak dilakukan proses lanjutan. Semua objek yang didapat selanjutnya digunakan untuk diekstrak fiturnya.

Pada tahap ekstraksi fitur digunakan untuk mengambil fitur atau ciri-ciri yang ada dari setiap citra, dimana fitur tersebut nantinya akan digunakan pada tahap klasifikasi. Terdapat 2 fitur yang akan diambil dari setiap citra, yang pertama adalah rata-rata jarak antar tiap titik gigitan dan yang kedua adalah jumlah titik gigitan yang ada pada citra tersebut. Nilai dari kedua fitur tersebut selanjutnya dinormalisasi dimana nilai berkisar antara 0 sampai 1 agar mempermudah dalam proses klasifikasi.

Teknik perhitungan rata-rata jarak (*Distance Metrics*) yang digunakan pada penelitian ini adalah *Euclidean Distance* dengan persamaan perhitungan jarak yang ditunjukkan pada persamaan (1). Metode yang digunakan dalam menormalisasi nilai adalah *Min-Max Normalization*. metode normalisasi ini akan melakukan transformasi linear terhadap data asli sehingga akan menghasilkan keseimbangan antar data [17], persamaan normalisasi ini ditunjukkan pada persamaan (2).

$$d(x, y) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2} \quad (1)$$

Dimana:

- $d(x, y)$  : Nilai jarak data  $x$  terhadap data  $y$ .
- $x$  : Vektor data  $x$ .
- $y$  : Vektor data  $y$ .
- $n$  : Jumlah titik gigitan ular

$$x_{new} = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (2)$$

Dimana:

- $x_{new}$  : Nilai hasil normalisasi.
- $x$  : Nilai yang akan dinormalisasi.
- $\min(x)$  : Nilai terendah pada dataset  $x$ .
- $\max(x)$  : Nilai tertinggi pada dataset  $x$ .

Tahap selanjutnya adalah klasifikasi menggunakan metode *Support Vector Machine* (SVM). SVM merupakan algoritma *supervised* yang merupakan algoritma pembelajaran secara terarah dengan melihat kelas



atau label dari data [15]. Kalsifikasi pada tahap ini akan menggunakan kernel-kernel SVM. Kernel SVM yang dipakai adalah RBF pada persamaan (3), *Polynomial* pada persamaan (4), dan *Linear* pada persamaan (5).

$$K(x_1, x_2) = \exp\left(-\frac{\|x_1 - x_2\|^2}{2\sigma^2}\right) \quad (3)$$

$$K(x_1, x_2) = (x_1^T x_2 + 1)^\rho \quad (4)$$

$$K(x_1, x_2) = (x_1^T x_2) \quad (5)$$

Dimana:

- $x_1, x_2$  : fitur dari vektor.
- $\sigma$  : Parameter bebas dengan  $\sigma \geq 0$ .
- $\rho$  : Parameter bebas dengan  $\rho \geq 0$ .

Dilihat dari persamaan kernel terdapat parameter  $\sigma$  (sigma) pada kernel RBF (3) dan parameter  $\rho$  pada kernel *Polynomial* (4). Parameter ini akan berpengaruh dalam proses penentuan *hyperplane* (garis pemisah) sehingga akan diuji berapa nilai terbaik untuk setiap parameter. Setiap pengujian nilai parameter akan dilakukan pengujian menggunakan data test dari model klasifikasi yang dibuat untuk memutuskan nilai parameter terbaik dilihat dari akurasi.

Pengukuran performansi sistem didapat dengan membangun *Confusion Matrix* untuk mencatat klasifikasi yang benar dan tidak benar bagi masing-masing kelas. Performansi dari sistem tidak hanya dilihat dari akurasi saja, dengan penggunaan *Confusion Matrix* selanjutnya dapat diketahui nilai-nilai kelas yang sudah diuji. Dengan memanfaatkan *Confusion Matrix* kemudian dapat dihitung nilai sensitivitas menggunakan persamaan (6), nilai spesifisitas pada persamaan (7), nilai akurasi pada persamaan (8) dan terakhir dihitung nilai *F1 Score* pada persamaan (11). Untuk mendapatkan nilai *F1 Score*, terlebih dahulu dicari nilai *precision* (10).

$$\text{Sensitivitas} = \frac{tp}{tp+fn} \quad (6)$$

$$\text{Spesifisitas} = \frac{tn}{tn+fp} \quad (7)$$

$$\text{Akurasi} = \frac{tp+tn}{tp+fn+tn+fp} \quad (8)$$

$$\text{Precision} = \frac{tp}{tp+fp} \quad (9)$$

$$\text{F1 Score} = 2 \times \frac{\text{precision} \times \text{sensitivitas}}{\text{precision} + \text{sensitivitas}} \quad (10)$$

Dimana:

- $tp$  (*True Positive*) : Gigitan ular berbisa yang benar teridentifikasi sebagai Gigitan ular berbisa.
- $tn$  (*True Negative*) : Gigitan ular tidak berbisa yang benar teridentifikasi sebagai Gigitan ular tidak berbisa.
- $fp$  (*False Positive*) : Gigitan ular tidak berbisa yang salah teridentifikasi sebagai Gigitan ular berbisa.
- $fn$  (*False Negative*): Gigitan ular berbisa yang salah teridentifikasi sebagai Gigitan ular tidak berbisa.

Nilai sensitivitas menunjukkan proporsi nilai dari kasus bernilai positif yang diprediksi benar positif, sedangkan spesifisitas merupakan kebalikan dari sensitivitas yaitu menunjukkan proporsi nilai dari kasus bernilai negatif yang dapat diprediksi benar negatif [18]. Pada penelitian ini nilai sensitivitas mewakili kemampuan sistem dalam mengkategorikan citra sebagai gigitan ular berbisa, sedangkan nilai spesifisitas mewakili kemampuan sistem dalam mengidentifikasi citra sebagai gigitan ular tidak berbisa. Nilai akurasi dihitung sebagai nilai evaluasi kinerja sistem secara keseluruhan, yaitu menghitung jumlah data yang dapat teridentifikasi dengan benar dan dibandingkan dengan total data. Semakin tinggi nilai akurasi menunjukkan bahwa sistem yang dibuat memiliki performa yang baik, begitu juga dengan nilai *F1 Score*.

### 3. ANALISA DAN PEMBAHASAN

Pada tahap ini dilakukan beberapa skenario pengujian. Hasil pengujian yang dilakukan meliputi pengaruh kernel-kernel SVM yang digunakan untuk pengklasifikasian citra, serta pengaruh nilai parameter kernel SVM pada akurasi yang dihasilkan. Skenario pengujian yang pertama adalah mencari nilai optimal parameter  $\sigma$  pada kernel RBF. Nilai  $\sigma$  yang diuji bernilai 0.1 sampai dengan 1. Skenario pengujian yang kedua adalah mencari nilai optimal parameter  $\rho$  pada kernel *Polynomial*. Nilai  $\rho$  yang diuji bernilai 0.1 sampai dengan 1. Dan pengujian terakhir adalah membandingkan kernel SVM, yaitu kernel RBF, kernel *Polynomial*, dan kernel *Linear*.

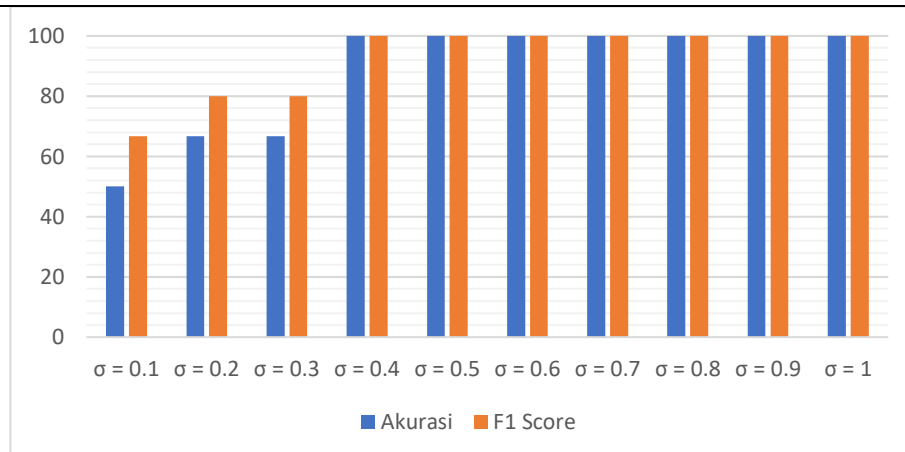


### 3.1 Pengujian Pertama

Skenario pengujian yang pertama adalah menguji nilai  $\sigma$  pada kernel RBF. Berdasarkan hasil pengujian pada Gambar 2, akurasi tertinggi yang didapat yaitu 100% pada saat rentang nilai  $\sigma$  pada 0.4 sampai dengan 1 dengan nilai *F1 Score* sebesar 100%. Sedangkan untuk akurasi terendah pada saat  $\sigma$  bernilai 0.1 dengan nilai akurasi sebesar 50% dan *F1 Score* sebesar 66.667%. Bisa diambil kesimpulan bahwa  $\sigma$  dengan nilai 0.4 memiliki akurasi tertinggi, karena memberikan nilai akurasi dan *F1 Score* tertinggi pertama. Berdasarkan hasil pengujian  $\sigma$  tersebut dapat dilihat bahwa semakin besar nilai  $\sigma$  dapat mempengaruhi akurasi sehingga akurasi lebih baik. Akurasi yang didapat tidak menentu berdasarkan karakteristik data set itu sendiri. Nilai performansi skenario pertama dapat dilihat pada Tabel 1.

**Tabel 1.** Hasil pengujian  $\sigma$  pada kernel RBF

Sigma	Sensitivitas	Spesitifitas	Akurasi	<i>F1 Score</i>
$\sigma = 0.1$	60%	0%	50%	66.667%
$\sigma = 0.2$	66.667%	0%	66.667%	80%
$\sigma = 0.3$	66.667%	0%	66.667%	80%
$\sigma = 0.4$	100%	100%	100%	100%
$\sigma = 0.5$	100%	100%	100%	100%
$\sigma = 0.6$	100%	100%	100%	100%
$\sigma = 0.7$	100%	100%	100%	100%
$\sigma = 0.8$	100%	100%	100%	100%
$\sigma = 0.9$	100%	100%	100%	100%
$\sigma = 1$	100%	100%	100%	100%



**Gambar 2.** Akurasi dan *F1 Score* pada pengujian  $\sigma$

Berdasarkan Gambar 2 ditunjukkan bahwa pengujian mampu mencapai akurasi dan *F1 Score* maksimum yaitu sebesar 100%. Sedangkan nilai terendah yang didapat yaitu nilai akurasi sebesar 50% dan *F1 Score* sebesar 66.667%. Nilai  $\sigma$  pada kernel RBF semakin besar cenderung membuat area *decision boundary* yang besar, begitu juga dengan semakin kecil nilai  $\sigma$  maka area *decision boundary* akan semakin kecil. Pesebaran data berpengaruh pada akurasi kernel ini.

### 3.2 Pengujian Kedua

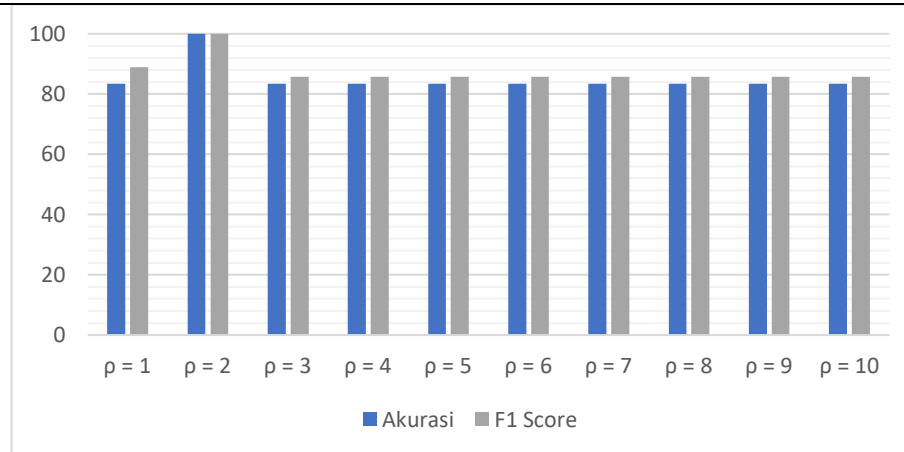
Skenario pengujian yang kedua adalah mencari nilai optimal parameter  $\rho$  (*polynom order / degree*) pada kernel *Polynomial*. Hasil pengujian skenario ini dapat dilihat pada Tabel 2, dan akurasi serta *F1 Score* pada Gambar 3. Nilai yang diuji yaitu antara 1 sampai dengan 10. Akurasi tertinggi yang didapat oleh data yaitu pada saat  $\rho = 2$  dengan nilai akurasi dan *F1 Score* sebesar 100%. Selain itu nilai terendah yang didapat adalah 83.333% untuk akurasi dan 85.714% untuk *F1 Score*. Dilihat pada Gambar 4, semakin besar  $\rho$  maka nilai akurasi dan *F1 Score* cenderung stabil seperti pada saat  $\rho = 3$  sampai  $\rho = 10$  nilai akurasi dan *F1 Score* cenderung sama dan tetap berada pada 83.333% dan 85.714%.

Grafik pada Gambar 3 menunjukkan bahwa nilai akurasi dan *F1 Score* mampu mencapai 100%, sedangkan nilai terendah yang didapat adalah 83.333% untuk akurasi dan 85.714% untuk *F1 Score*. Nilai  $\rho$  pada kernel *Polynomial* berpengaruh pada bentuk dari *decision boundary*. Nilai  $\rho$  yang lebih besar membuat *decision boundary* semakin fleksibel, sebaliknya jika nilai lebih kecil maka *decision boundary* cenderung lurus.



**Tabel 2.** Hasil pengujian  $\rho$  pada kernel Polynomial

Degree	Sensitivitas	Spesitifitas	Akurasi	<i>F1 Score</i>
$\rho = 1$	80%	100%	83.333%	88.889%
$\rho = 2$	100%	100%	100%	100%
$\rho = 3$	100%	66.667%	83.333%	85.714%
$\rho = 4$	100%	66.667%	83.333%	85.714%
$\rho = 5$	100%	66.667%	83.333%	85.714%
$\rho = 6$	100%	66.667%	83.333%	85.714%
$\rho = 7$	100%	66.667%	83.333%	85.714%
$\rho = 8$	100%	66.667%	83.333%	85.714%
$\rho = 9$	100%	66.667%	83.333%	85.714%
$\rho = 10$	100%	66.667%	83.333%	85.714%



**Gambar 3.** Akurasi dan F1 Score pada pengujian  $\rho$

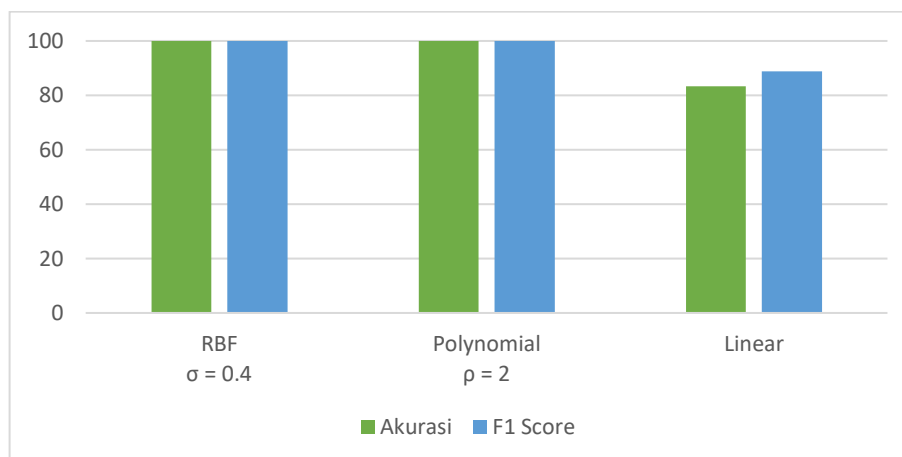
### 3.3 Pengujian Ketiga

Skenario pengujian yang ketiga adalah membandingkan akurasi setiap kernel SVM. Kernel yang dipakai pada skenario pengujian ini adalah RBF, *Polynomial*, dan ditambah kernel *Linear*. Nilai optimal  $\sigma$  untuk kernel RBF yang didapat pada skenario pengujian pertama dipakai pada skenario ini. Nilai optimal  $\sigma$  yang didapat pada skenario pengujian pertama adalah 0.4. Begitu juga dengan nilai optimal  $\rho$  yang didapat pada skenario pengujian kedua digunakan untuk kernel *Polynomial*. Nilai optimal yang didapat  $\rho$  pada skenario pengujian kedua yaitu 2. Nilai performansi skenario ketiga dapat dilihat pada Tabel 3.

Perbandingan akurasi setiap kernel dapat dilihat pada Gambar 4. Kernel RBF dan *Polynomial* mampu mengklasifikasikan data dengan sangat baik dan mendapat nilai akurasi optimal sebesar 100%. Nilai sensitivitas dan spesifisitas pada kernel tersebut juga mendapat nilai optimal yaitu 100%, yang berarti semua data *test* dapat dipisahkan sesuai labelnya dengan benar. Hanya kernel *Linear* saja pada yang tidak menghasilkan akurasi sempurna dengan nilai sensitivitas, spesifisitas, akurasi, dan *F1 Score* untuk masing masing nilai sebesar 80%, 100%, 83.333%, dan 88.889%. Hal ini berarti bahwa hanya 1 dari 2 data *test* yang mampu dikenali sebagai gigitan ular tidak berbisa, dan 4 dari 4 data *test* yang mampu dikenali sebagai gigitan ular berbisa, dan diketahui akurasi yang diperoleh sebesar 83.333%, dimana hanya 1 dari 6 data yang salah diidentifikasi. Rata-rata setiap kernel sangat baik dimana performansi didapat diatas 90%. Hal ini bisa disebabkan karena pada tahap *preprocessing* dilakukan proses lebih lanjut agar pengeksraksian fitur dapat dilakukan lebih maksimal dan proses klasifikasi akan jauh lebih baik. Selain itu, pesebaran data juga mempengaruhi akurasi dari klasifikasi. Pesebaran data yang teratur dan terpisah dengan baik membantu proses klasifikasi.

**Tabel 3.** Hasil pengujian kernel SVM

Degree	Sensitivitas	Spesitifitas	Akurasi	<i>F1 Score</i>
RBF $\sigma = 0.4$	100%	100%	100%	100%
Polynomial $\rho = 2$	100%	100%	100%	100%
Linear	80%	100%	83.333%	88.889%
Rata-rata	93.333%	100%	94.444%	96.296%



**Gambar 4.** Akurasi dan *F1 Score* kernel SVM

Dilihat pada Gambar 4, hasil pengujian memiliki tingkat akurasi dan *F1 Score* tertinggi sebesar 100% pada kernel RBF dengan  $\sigma = 0.4$  dan *Polynomial* dengan  $\rho = 2$ . Sedangkan untuk kernel linear, akurasi yang didapat sebesar 83.333% dan *F1 Score* sebesar 88.889%.

### 3.4 Active Contour Model

Untuk penggunaan *Active Contour Model* tidak terlalu memberikan pengaruh *filtering* pada citra. Iterasi yang digunakan pada proses *Active Contour Model* tidak memberikan hasil yang maksimal. Semakin besar nilai iterasi tidak memberikan jaminan titik gigitan ular pada citra tersegmentasi dengan baik. Hal ini dikarenakan *Active Contour Model* merupakan metode yang baik untuk mengsegmentasi objek tunggal. Tetapi penggunaan metode *Active Contour Model* masih dapat digunakan pada penelitian ini.

## 4. KESIMPULAN

Berdasarkan hasil skenario pengujian dan analisis yang telah dilakukan dapat disimpulkan bahwa sistem dapat mendeteksi gigitan ular berbisa dan gigitan ular tidak berbisa. Sistem ini memiliki performansi yang cukup baik dan dapat mendeteksi dengan tepat. Sistem ini mendapat nilai sensitivitas sebesar 100%, nilai spesifitas sebesar 100%, nilai akurasi sebesar 100%, dan nilai *F1 Score* sebesar 100% pada kernel RBF dan *Polynomial*. Penggunaan metode *Active Contour Model* tidak terlalu banyak membantu untuk proses ekstraksi fitur citra. Hal yang utama dari proses klasifikasi adalah pada tahap preprocessing, jika tahap ini dapat diidentifikasi dengan baik maka fitur yang didapat antar data akan maksimal sehingga proses klasifikasi akan terbantu dan akurasi yang didapat akan tinggi.

Walaupun memiliki akurasi 100%, namun sistem ini belum dapat dikatakan optimal dan sempurna dikarenakan data yang digunakan pada penelitian ini hanya berjumlah 20 data. Data yang didapat pun memiliki kualitas yang jauh dari kata bagus, seperti kurangnya pencahayaan pada citra, banyaknya noise pada citra, resolusi yang kecil, dan minimnya objek yang ada pada citra. Untuk penelitian selanjutnya diperlukan data yang lebih memadai agar dapat mempermudah tahap preprocessing sampai klasifikasi dan dapat meningkatkan performansi sistem.

## REFERENCES

- [1] juragandavin. Mendefinisikan ciri-ciri ular berbisa. 18 April 2016. <http://www.juragandavin.com/mendefinisikan-ciri-ciri-ular-berbisa>. [Diakses 8 Oktober 2018].
- [2] WHO. Snakebite envenoming. 20 Februari 2018. <http://www.who.int/news-room/fact-sheets/detail/snakebite-envenoming>. [Diakses 9 Oktober 2018].
- [3] LIPI. Pendataan Kasus Gigitan Ular Berbisa Terabaikan. 26 Oktober 2016. <http://lipi.go.id/lipimedia/pendataan-kasus-gigitan-ular-berbisa-terabaikan/16730>. [Diakses 8 Oktober 2018].
- [4] Williams, Donna J., and Shah, Mubarak. (1991). A Fast Algorithm for Active Contours and Curvature Estimation. Jurnal. Florida: Department of Computer Science, University of Florida, Orlando.
- [5] Cai, Y.D., Ricardo, P.W., Jen, C.H. and Chou, K.C., (2004). Application of SVM to predict membrane protein types. *Journal of Theoretical Biology*, 226(4), pp.373-376.
- [6] Budianto, Aris. (2019). Perbandingan K-Nearest Neighbor (KNN) Dan Support Vector Machine (SVM) Dalam Pengenalan Karakter Plat Kendaraan Bermotor. 10.20961/jiptek.v11i1.18018.
- [7] Putra, Rayiemas Manggala., Adiwijaya., Utama, Dody Qori. (2019). Snake bite classification using Chain code and K nearest neighbour. *Journal of Physics: Conference Series*. 1192. 012015. 10.1088/1742-6596/1192/1/012015.
- [8] Z, H. Pradana. (2016). Beef Cattle Weight Determine by Using Digital Image Processing. pp. 13-16.



- [9] M. Haralick, Robert. 1973. Textural Features for Image Classification. IEEE.USA.
- [10] Soelaiman, R., Agustianty, S., Purwananto, Y., & Purnama, I. E. (2012). Jurnal Ilmu Komputer Dan Informasi, 2(1), 25.
- [11] Prijono, Agus., Wijaya, Marvin Ch. (2007). Pengolahan Citra Digital Menggunakan MatLAB Image Processing Toolbox. Bandung : Informatika.
- [12] Duda, Richard O., Hart, Peter E. (1973). Pattern Classification and Scene Analyst. Wiley.
- [13] Budiharto, Widodo. (2016). Machine Learning and Computational Intelligence, Yogyakarta: CV. ANDI OFFSET.
- [14] Nugroho, Anto Satriyo., Witarto, Arief Budi., Handoko, Dwi. (2003). Support Vector Machine Teori dan Aplikasinya dalam Bioinformatika. IlmuKomputer.Com.
- [15] Muis, I. A., Affandes, M., Muis, I. A., P. Studi, T. Informatika, and F. Sains. (2015). Penerapan Metode Support Vector Machine (SVM) Menggunakan Kernel Radial Basis Function (RBF) Pada Klasifikasi Tweet. J. Sains, Teknol. dan Ind., vol. 12, no. 2, pp. 189–197.
- [16] Cristianini, N. and Shawe-Taylor, J., (2000). An introduction to support vector machines and other kernel-based learning methods. Cambridge university press.
- [17] Nishioka, S. D. A., Silveira, P. V. P., and Bauab, F. A. (1995). Bite marks are useful for the differential diagnosis of snakebite in brazil. Wilderness & environmental medicine, 6(2):183–188.
- [18] Powers, D. M. (2011). Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation.