

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Tebu

Tanaman semipermanen dengan nilai sosial ekonomi yang signifikan, tebu (*Saccharum officinarum*) ditanam di seluruh wilayah tropis dan subtropis di dunia. Lebih dari 70% dari produksi gula dunia berasal dari tebu, yang merupakan tanaman yang berguna. Selulosa dan *hemicellulose* yang ditemukan dalam sisa-sisa tebu dapat digunakan untuk membuat bioetanol dan biofuel transportasi cair lainnya. Biofuel yang terbuat dari tebu memiliki potensi untuk mengurangi polusi gas rumah kaca (GHG) yang disebabkan oleh pembakaran bahan bakar fosil. Di Brazil, misalnya, bioetanol gandum dapat mengurangi polusi gas rumah kaca dari bahan bakar fosil sebesar 85%. Dalam beberapa dekade terakhir, tebu telah menyebar dengan cepat, mengalihkan produk lain (seperti kedelai, beras, dan jagung), rumput, dan hutan. Di satu sisi, pertanian dan keamanan pangan dapat terancam jika tebu menggantikan produk makanan lainnya. Namun, dengan mengubah albedo permukaan dan evapotranspirasi, pertumbuhan tebu dapat memiliki dampak pada iklim lokal. Untuk tujuan mengelola produksi tebu dan memastikan produksi tebu yang berkelanjutan, perkiraan tepat waktu dan akurat dari area yang ditanam tebu diperlukan untuk melacak kondisi pertumbuhan dan perkiraan hasil [6].

Pendeteksian pohon tebu telah dilakukan sebelumnya oleh Zheng et al. (tahun 2022) menggunakan metode TWDTW (*Time-Weighted Dynamic Time Warping*) atau tergolong metode berbasis fenologi dengan citra Landsat-7/8, Sentinel-1 dan Sentinel-2. Metode yang digunakan dalam penelitian ini berhasil melakukan identifikasi area budidaya tebu secara efektif dan akurat walaupun dengan data sampel pelatihan yang terbatas. Peta tebu yang dihasilkan dapat diterapkan

untuk pemantauan kondisi pertumbuhan dan hasil tebu, serta berkontribusi pada produksi tebu berkelanjutan. Selain melakukan pembatasan dengan menggunakan pencampuran spektral jenis vegetasi lain di area tanaman yang heterogen, keakuratan pemetaan juga dapat dilakukan dengan membandingkan secara langsung dengan metode lain, seperti *Machine Learning*.

2.1.2 *Machine Learning*

Machine Learning adalah pendekatan *Artificial Intelligence* yang berfokus pada pembuatan mesin (robot) yang dapat belajar tanpa diprogram secara eksplisit (detail). *Machine Learning* adalah teknik untuk melakukan inferensi terhadap data dengan pendekatan matematis (turunan dari matematika dan statistika). Maksud dari inferensi tersebut adalah lebih memfokuskan relasi antar atribut. Intinya, *Machine Learning* digunakan untuk membuat model (matematis) yang merefleksikan pola – pola data. *Machine Learning* dapat diibaratkan menjadi sebuah alat yang identik dengan rumus matematika yang cara menggunakannya bergantung pada domain permasalahan. Terdapat 2 metode pembelajaran yang banyak digunakan pada *Machine Learning*, yaitu pembelajaran terawasi (*supervised learning*) dan pembelajaran tidak terawasi (*unsupervised learning*) [7].

2.1.3 *Supervised Learning*

Supervised Learning adalah *Machine Learning* model yang mempelajari data dengan menggunakan data label atau target. Pada supervised learning, model akan membutuhkan data training berupa input dan target data. Model ini nantinya akan dilatih untuk dapat melakukan prediksi berdasarkan pola yang ditemukan dalam menjawab data target, yang kemudian dievaluasi dan dibandingkan hasilnya dengan prediksi oleh *test data*. Contoh metode dari model ini adalah klasifikasi dan regresi.

2.1.4 Klasifikasi

Klasifikasi adalah suatu proses penemuan kumpulan model atau fungsi yang menjelaskan dan membedakan data kedalam kelas tertentu. Tujuannya adalah agar model tersebut dapat digunakan dalam penentuan kelas dari suatu objek yang belum diketahui kelasnya. Proses klasifikasi terjadi dalam 2 proses, yaitu :

1. Proses *learning / training*, membangun model menggunakan data *training*.
2. Proses *testing, testing* data menggunakan model yang telah didapat dari proses *training*.

Contoh algoritma yang sering digunakan untuk klasifikasi, yaitu *Linier Regression, Logistic Regression, Random Forest, XGBoost, K-NN* dan *SVM*.

2.1.5 *Support Vector Machine* (SVM)

Pertama kali diperkenalkan oleh Vapnik pada tahun 1992 sebagai rangkaian harmonis konsep – konsep unggulan dalam bidang *pattern recognition* (pengenalan pola). Usia SVM masih terbilang muda sebagai salah satu metode pengenalan pola. Walaupun begitu, evaluasi kemampuan dalam berbagai aplikasi menempatkannya sebagai karya terbaik dalam pengenalan pola. *Support Vector Machine* adalah metode *Machine Learning* yang bekerja atas prinsip *Structural Risk Minimization (SRM)* dengan tujuan menemukan *hyperplane* terbaik yang memisahkan 2 kelas (*class -1* dan *+1*) pada *input space*[8]. Menemukan *hyperplane* terbaik dapat dilakukan dengan menghitung nilai *margin hyperplane* tersebut, yaitu dengan pencarian titik maksimalnya. *Margin* merupakan jarak antara *hyperplane* dengan *pattern* terdekat (*support vector*) dari setiap kelas. Diasumsikan kedua kelas dapat terpisah secara sempurna, sehingga diperoleh persamaan (1)[9].

$$\mathbf{w} \cdot \mathbf{x}_i + \mathbf{b} = 0(1)$$

Bidang pemisah yang dapat memisahkan *dataset* yang sesuai dengan *class* dan memiliki *margin* paling besar merupakan bidang pemisah terbaik. Bidang pemisah pertama adalah untuk membatasi *class* pertama dan bidang pemisah kedua adalah untuk membatasi *class* kedua, yang didefinisikan dengan persamaan (2) dan persamaan (3)[9].

$$(\mathbf{w} \cdot \mathbf{x}_i + \mathbf{b}) \geq +1 \text{ untuk } \mathbf{y}_i = +1(2)$$

$$(\mathbf{w} \cdot \mathbf{x}_i + \mathbf{b}) \geq -1 \text{ untuk } \mathbf{y}_i = -1(3)$$

Variabel \mathbf{w} adalah bidang normal antara bidang pemisah terhadap pusat koordinat, variabel \mathbf{b} merupakan posisi bidang relatif terhadap pusat koordinat, dan jarak antara *hyperplane* dan titik terdekat untuk menemukan margin terbesar adalah dengan memaksimalkan fungsi $\frac{1}{\|\mathbf{w}\|}$ atau dengan meminimalkan $\|\mathbf{w}\|^2$. Untuk bidang pemisah atau *hyperplane* kedua kelas dapat dijelaskan dengan persamaan (4) atau biasa disebut Quadratic Programming[9].

$$\mathbf{y}_i (\mathbf{w} \cdot \mathbf{x}_i + \mathbf{b}) \geq 1, i = 1, 2, 3, \dots, n(4)$$

Konsep dasar dari SVM merupakan kombinasi harmonis dari teori – teori komputasi yang telah ada puluhan tahun sebelumnya, seperti *margin hyperplane*, *kernel* yang diperkenalkan oleh Aronszajn (tahun 1950) dan demikian juga dengan konsep – konsep pendukung yang lain. Berbeda dengan strategi *neural network* yang berusaha mencari *hyperplane* pemisah antar kelas, SVM berusaha menemukan *hyperplane* terbaik pada *input space*[8].

Berikut ini adalah beberapa karakteristik dari algoritma *Support Vector Machine* (SVM).

- Prinsip dasar dari SVM adalah *linier classifier*, selanjutnya akan dikembangkan agar dapat bekerja pada problem *non-linier* dengan memasukkan konsep *kernel trick* pada ruang kerja berdimensi tinggi.
- *Pattern recognition* dilakukan dengan mentransformasikan data pada *input space* ke ruang yang dimensi lebih tinggi dan optimasi dilakukan di ruang vector yang baru tersebut. Perbedaan SVM dari solusi *pattern recognition* pada umumnya adalah melakukan optimasi parameter pada ruang hasil transformasi yang berdimensi lebih rendah dari dimensi *input space*.
- Strategi SVM adalah *Structural Risk Minimization* (SRM).
- Prinsip kerjanya hanya mampu melakukan klasifikasi 2 kelas.

Support Vector Machine telah terbukti sukses diaplikasikan dalam menyelesaikan masalah klasifikasi dan estimasi fungsi setelah pengenalan yang dilakukan oleh Vapnik dalam konteks teori *statistical learning* dan *structural risk minimization*. Vapnik mengkonstruksikan SVM standar untuk memisahkan data – data pelatihan menjadi 2 kelas [8].

Kelebihan dan kekurangan dari *Support Vector Machine* adalah sebagai berikut.

Kelebihan SVM :

- Generalisasi, sebagai kemampuan suatu metode untuk mengklasifikasikan suatu *pattern*, yang tidak termasuk data yang dipakai dalam fase pembelajaran metode tersebut. *Generalization error* dipengaruhi oleh 2 faktor, yaitu *training set* dan dimensi VC (*Vapnik-Chervonenkis*). Namun SVM dapat meminimalkan error pada kedua faktor tersebut.
- *Curse of Dimensionality*, sebagai masalah yang dihadapi suatu metode *pattern recognition* dalam melakukan estimasi parameter (contoh : jumlah hidden neuron pada neural network, stopping

criteria pada proses pembelajaran) dikarenakan jumlah sampel data yang relatif sedikit dibandingkan dimensional ruang vektor data tersebut. Semakin tinggi dimensi dari ruang vektor informasi yang diolah, membawa konsekuensi dibutuhkan jumlah data dalam proses pembelajaran.

- *Feasibility*, mengimplementasikan SVM relatif mudah, karena proses penentuan *support vector* dapat dirumuskan dalam *Quadratic Programming Problem*. Sehingga kita memiliki *library* untuk menyelesaikannya menggunakan SVM dengan mudah [8].

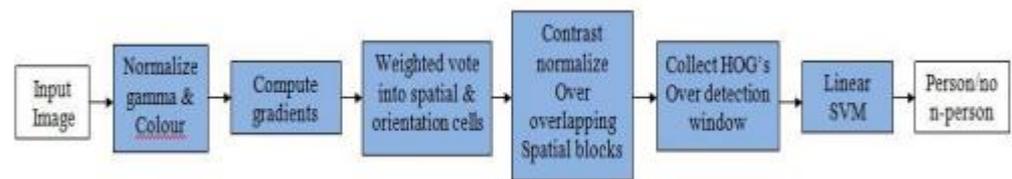
Kekurangan SVM :

- Sulit digunakan dalam permasalahan berskala besar.
- SVM secara teoritik dikembangkan untuk permasalahan klasifikasi dengan 2 kelas atau lebih. Namun, setiap strategi ini memiliki kelemahan, sehingga dapat dikatakan penelitian dan pengembangan SVM pada *multiclass problem* masih menjadi tema penelitian terbuka [8].

2.1.6 *Histogram of Oriented Gradients (HOG)*

Histogram of oriented gradients atau biasa disingkat HOG adalah suatu fitur yang umum digunakan sebagai algoritma ekstraksi dalam computer vision dan pemrosesan gambar. Pertama kali diperkenalkan oleh Dalal dan Triggs pada tahun 2005 pada pendeteksian manusia. Berbagai masalah pendeteksian objek telah diselesaikan menggunakan algoritma ini, seperti deteksi pengenalan dan pelacakan pejalan kaki, deteksi pengenalan gerakan tangan, deteksi pengenalan wajah, deteksi mata dan pengenalan bagian tubuh untuk pelacakan, deteksi mobil, deteksi hewan maupun deteksi buah. HOG telah terbukti efektif untuk klasifikasi dan memberikan akurasi deteksi yang tinggi dengan perhitungan yang sederhana. Fitur HOG ini mampu memberikan gambaran informasi tentang bentuk dan tampilan objek lokal dengan distribusi gradien intensitas lokal atau arah tepi, hal tersebut karena fitur

ini memiliki ketahanan yang terhadap perubahan iluminasi dan bayangan. *Histogram of Oriented Gradients* adalah suatu metode yang digunakan untuk melakukan deteksi keberadaan objek sebelum terdeteksi oleh kamera. Ada beberapa tahapan yang dilakukan pada metode ini [5].



Gambar 2.1 Diagram Alir HOG

Umumnya, pendeteksian objek menggunakan algoritma HOG dilakukan dalam 3 tahap, yaitu komputasi gradien, binning orientasi dan normalisasi block. Langkah awalnya adalah melakukan konversi citra RGB (*Red, Green, Blue*) menjadi *grayscale* yang dilakukan dengan perhitungan sebagai berikut pada persamaan (5).

$$f_0(x,y) = \frac{f_i^R(x,y) + f_i^G(x,y) + f_i^B(x,y)}{3} \quad (5)$$

Kemudian selanjutnya melakukan perhitungan nilai gradien setiap piksel. Setelah nilai gradien didapatkan, proses berikutnya adalah menentukan jumlah bin orientasi yang akan digunakan dalam pembuatan histogram (*orientation binning*). Namun sebelumnya, pada proses komputasi gradien (*gradient compute*), gambar pelatihan dibagi menjadi beberapa *cell* dan dikelompokkan menjadi ukuran lebih besar (*block*). Kemudian pada proses normalisasi block, digunakan perhitungan geometri R-HOG, ini dilakukan karena terdapat block yang saling tumpang tindih. Hal ini berbeda dengan proses pembuatan histogram citra yang menggunakan nilai – nilai intensitas piksel dari

suatu citra atau bagian tertentu dari citra untuk pembuatan histogram[10].

Proses normalisasi blok ini memiliki beberapa langkah sebagai berikut yang dijabarkan pada persamaan (6), (7) dan (8).

$$L2 - norm: v \rightarrow v = \frac{v}{\sqrt{\|v\|_k^2 + \epsilon^2}} \quad (6)$$

L2 - hys: L2 - norm diikuti oleh pemotongan dan re-normalisasi

$$L1 - norm: v \rightarrow v = \frac{v}{\|v\|_k + \epsilon} \quad (7)$$

$$L1 - sqrt: v \rightarrow v = \sqrt{\frac{v}{\|v\|_k + \epsilon}} \quad (8)$$

Variabel v mewakili vektor yang tidak dinormalisasi yang berisi seluruh elemen histogram dalam satu blok, variabel $\|v\|_k$ adalah k-norm untuk $k = 1, 2$ dan variabel ϵ adalah nilai konstanta kecil untuk menghindari pembagian dengan nol. Hasil akhir vektor fitur HOG diperoleh dengan menggabungkan semua komponen yang dinormalisasi dari semua blok di jendela deteksi menjadi suatu vektor besar. Ukuran vektor total harus $N \times (C \times C) \times (B \times B)$ dengan N adalah jumlah bin orientasi, $(C \times C)$ adalah dimensi sel dan $(B \times B)$ adalah dimensi blok di jendela deteksi [5].

2.2 Penelitian Terdahulu

Tabel 2.1 Penelitian Terdahulu

No.	Judul Jurnal	Pengarang	Algoritma	Kekurangan
1.	Development of Phenology-Based for Identifying Sugarcane Plantation Areas in China Using High-Resolution	Yi Zheng, Xiangqian Li, dan Wen Ping Yuan (Tahun 2022)	TWDTW, NDVI, VH (pemisah antara pisang dan tebu), PA, UA, OA (akurasi tingkat pixel), R2, RMSE,	1.) Metode TWDTW kurang sensitive terhadap sampel pelatihan dibandingkan Machine Learning, sehingga dapat

	Satellite Datasets[6]		RMAE (akurasi tingkat regional)	<p>dengan mudah diterapkan pada skala besar dan tahun berbeda dengan sampel terbatas.</p> <p>2.) Masih ada potensi ketidakpastian dalam metode ini, akan ada kemungkinan terjadinya efek salt-and-pepper di peta tebu karena kualitas gambar optik yang buruk. Oleh karena itu, yang dilakukan pada penelitian selanjutnya adalah menerapkan teknik segmentasi gambar berbasis objek atau pixel untuk mengurangi efek salt-and-pepper dan meningkatkan kinerja pemetaan di peta tebu.</p>
2.	Detecting and Counting	Yudhi Prabowo dan	Histogram of Oriented Gradient	1.) Waktu perolehan kumpulan gambar

	Coconut Trees in Pleiades Satellite Imagery Using Histogram of Oriented Gradients and Support Vector Machine[5]	Kenlo Nishida Nasahara (Tahun 2019)	(HOG) sebagai feature extractor dan Support Vector Machine (SVM) sebagai classifier	<p>lebih dari 3 tahun sehingga sudah banyak pohon kelapa yang ditebang dan diganti dengan bangunan, jalan dan kolam renang kecil.</p> <p>2.) Waktu komputasi sekitar 2 jam menggunakan bahasa Python murni sehingga terlalu lambat dan tidak efisien untuk tugas deteksi objek. Kemudian dioptimalkan menggunakan Cython untuk meningkatkan kecepatan perhitungan dan berhasil mencapai sekitar 2 menit (60x lebih cepat daripada kinerja Python murni).</p>
3.	Evaluation The Accuracy of Oil Palm Tree	Nafisah Khalid dan Nur Aina	Deep Learning, Support Vector Machine dan	1.) Deep Learning mendapatkan akurasi tertinggi

	Detection Using Deep Learning and Support Vector Machine Classifier	Shahrol (Tahun 2022)	ArcGIS (aplikasi penilaian akurasi)	<p>karena metode ini hanya mengklasifikasikan 1 pola dan pixel homogen tertentu untuk 1 objek.</p> <p>2.) SVM memiliki nilai akurasi pengguna dan produsen lebih tinggi daripada deep learning, akurasi pengguna sebesar 94% berarti ada lebih sedikit kesalahan komisi dan akurasi produsen sebesar 77% berarti ada lebih sedikit kesalahan penghilangan.</p>
4.	Automatic Detection of Individual Oil Palm Trees from UAV Images Using HOG Features and SVM Classifier	Yiran Wang, Xiaolin Zhu dan Bo Wu (Tahun 2018)	HOG untuk mengekstrak fitur yang mampu menggambarkan tekstur pohon palm, SVM untuk mengimplementasikan tugas klasifikasi biner	Dalam pengembangan selanjutnya, akurasi deteksi dapat ditingkatkan dengan memperbesar dataset pelatihan agar mendapatkan gambar yang lebih representatif.

			dan PA, UA, OA sebagai akurasi	
5.	<p>Deteksi Kebutuhan Nitrogen untuk Penentuan Jumlah Pupuk pada Tanaman Padi Berdasarkan Warna Daun Menggunakan Support Vector Machine</p>	<p>Gumilar Fajar Darajat dan Irfan Maliki (Tahun 2019)</p>	<p>Histogram of Oriented Gradient dan Support Vector Machine</p>	<p>Untuk mengurangi kekurangan dalam penelitian,</p> <p>1.) Dilakukan penambahan data dengan akuisisi citra yang lebihvariatif, baik dari ukuran, resolusi citra, jenis kamera, dan aspek teknis akuisisi lainnya sehingga data yang digunakan dapat menangani data uji yang bervariasi.</p> <p>2.) Untuk memperoleh data latih yang lebih presisi, sebaiknya pada saat akuisisi data citra selalu didampingi oleh peneliti yang berkompeten serta perhatikan kualitas daun padi yang akan dipilih.</p>