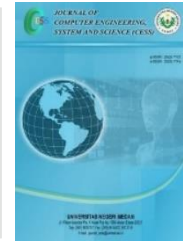


Contents list available at [www.jurnal.unimed.ac.id](http://www.jurnal.unimed.ac.id)

**CESS**  
**(Journal of Computing Engineering, System and Science)**

journal homepage: <https://jurnal.unimed.ac.id/2012/index.php/cess>



## Komparasi Kernel Support Vector Machine untuk Deteksi Tangga Turun dan Lantai

### Kernel Comparison on Support Vector Machine for Detecting Stairs Descent

Ahmad Wali Satria Bahari Johan<sup>1\*</sup>, Ardian Yusuf Wicaksono<sup>2</sup>, Muhammad Dzulfikar Fauzi<sup>3</sup>, Rizky Fenaldo Maulana<sup>4</sup>, Kharisma Monika Dian Pertiwi<sup>5</sup>

<sup>1,2,3,4,5</sup> Program Studi Informatika, Fakultas Teknologi Informasi dan Bisnis,  
Institut Teknologi Telkom Surabaya, Surabaya, Indonesia

email: <sup>1</sup>[ahmadsatria13@ittelkom-sby.ac.id](mailto:ahmadsatria13@ittelkom-sby.ac.id), <sup>2</sup>[ardian@ittelkom-sby.ac.id](mailto:ardian@ittelkom-sby.ac.id), <sup>3</sup>[muhammad.dzulfikar.f@ittelkom-sby.ac.id](mailto:muhammad.dzulfikar.f@ittelkom-sby.ac.id), <sup>4</sup>[rizkyfenaldo@ittelkom-sby.ac.id](mailto:rizkyfenaldo@ittelkom-sby.ac.id), <sup>5</sup>[kharismamonika@ittelkom-sby.ac.id](mailto:kharismamonika@ittelkom-sby.ac.id)

Submitted: 23 Maret 2022 | Revision: 06 Juni 2022 | Accepted: 15 Juni 2022

#### ABSTRAK

Terdapat 4 kernel yang dapat digunakan dalam klasifikasi *Support Vector Machine* dalam membuat *hyperplane*. Keempat kernel tersebut adalah *linear*, *polynomial*, *gaussian* dan *sigmoid*. Setiap kernel dapat menghasilkan akurasi yang berbeda-beda. Hal ini dikarenakan pengaruh sebaran data yang diklasifikasikan. Terdapat 2 kelas yang diklasifikasikan, yaitu lantai dan tangga turun. Dilakukan proses ekstraksi fitur tekstur terhadap citra lantai dan tangga turun menggunakan metode *Gray Level Co-occurrence Matrix*. Terdapat 7 fitur dari GLCM yang dihasilkan pada proses ekstraksi fitur. Selanjutnya dilakukan klasifikasi menggunakan *Support Vector Machine* dengan mencoba setiap kernelnya. Dari hasil pengujian didapatkan kernel *linear* menghasilkan akurasi yang paling tinggi, yaitu 89%. Kernel *sigmoid* mendapatkan akurasi 84%. Kernel *Gaussian* mendapatkan akurasi sebesar 85%. Sedangkan kernel *polynomial* mendapatkan akurasi yang paling rendah yaitu 78%.

**Kata Kunci:** *Support Vector Machine; Linear; Polynomial; Gaussian; Sigmoid.*

#### ABSTRACT

There are 4 kernels that can be used in the *Support Vector Machine* classification in creating hyperplanes. The four kernels are *linear*, *polynomial*, *gaussian*, and *sigmoid*. Each kernel can produce different accuracy. This is due to the influence of the distribution of classified data. There are 2 classes that are classified, namely floors and stairs descent. The texture feature extraction process is carried out on the image of the floor and stairs going down using the *Gray Level Co-occurrence Matrix* method. There are 7 features of GLCM generated in the

\*Penulis Korespondensi:

email: [ahmadsatria13@ittelkom-sby.ac.id](mailto:ahmadsatria13@ittelkom-sby.ac.id)

feature extraction process. Furthermore, classification is carried out using the Support Vector Machine by trying each kernel. From the test results, the linear kernel produces the highest accuracy, which is 89%. The sigmoid kernel gets 84% accuracy. Gaussian kernels get an accuracy of 85%. While the polynomial kernel gets the lowest accuracy, which is 78%.

**Keywords:** *Support Vector Machine, Linear, Polynomial, Gaussian, Sigmoid.*

---

## 1. PENDAHULUAN

Klasifikasi merupakan cara untuk mengkategorikan atau mengelompokkan objek yang telah dikenali berdasarkan dari data latih yang digunakan. Klasifikasi berguna untuk membantu seseorang dalam mengkategorikan objek-objek yang cukup banyak dengan waktu yang singkat. Terdapat 2 tipe klasifikasi yang dapat digunakan dalam mengkategorikan objek, yaitu *supervised learning* dan *unsupervised learning*[1]. *Supervised learning* merupakan algoritma klasifikasi yang melakukan pembelajaran terlebih dahulu sebelum melakukan pengujian klasifikasi terhadap data uji. Dalam proses pembelajaran akan menghasilkan bobot pada setiap label yang digunakan dalam pengujian terhadap data uji. Sedangkan *unsupervised learning* merupakan metode klasifikasi yang tidak melakukan proses pelatihan menggunakan data training. *Unsupervised learning* langsung melakukan pengujian dengan berlandaskan data latih untuk mengklasifikasikan data uji[1][2].

Algoritma *Support Vector Machine* (SVM) merupakan salah satu metode klasifikasi yang bertipe supervised learning. Kelebihan dari SVM yaitu metode ini dapat mengatasi masalah klasifikasi dengan cara linear maupun non linear[3][4]. SVM akan mencari hyperplane terbaik untuk memaksimalkan jarak antar kelas atau label yang diklasifikasikan. Dimana hyperplane memisahkan antar kelas yang akan diklasifikasikan. Namun, umumnya dataset yang digunakan tidak dapat dipisahkan secara linear. Oleh karena itu dibutuhkan kernel untuk mentransformasikan data ke ruang dimensi yang lebih tinggi yang disebut ruang kernel yang berguna untuk memisahkan data secara Linear[5]. Umumnya, terdapat 4 fungsi kernel pada algoritma SVM, yaitu *linear*, *Polynomial*, *Gaussian* dan *Sigmoid*. Setiap kernel dapat menghasilkan akurasi yang berbeda-beda. Hal ini dipengaruhi akan karakteristik data yang digunakan. Sehingga diperlukan percobaan masing-masing kernel untuk mengetahui kernel mana yang memberikan akurasi tertinggi untuk melakukan klasifikasi. Dalam penelitian ini, kami menggunakan data tangga turun dan lantai yang akan diklasifikasikan. Kami mencari akurasi terbaik dalam mendeteksi tangga turun. Akurasi yang bagus sangat diperlukan dalam mendeteksi tangga turun, dikarenakan nantinya sistem deteksi tangga turunan akan digunakan oleh orang buta dalam mendeteksi halangan tangga turun. Sehingga diperlukan ketepatan dalam hal deteksi.

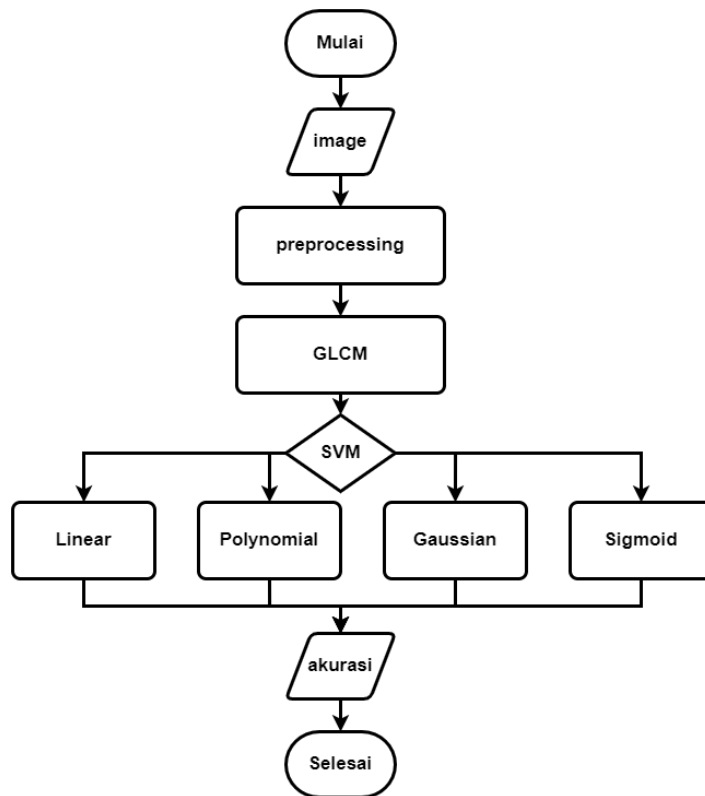
Beberapa penelitian terkait deteksi halangan telah dilakukan oleh beberapa peneliti. Penelitian yang dilakukan oleh AbdelkaderDairi[6]. Penelitian tersebut menggunakan metode Deep Stacked Autoencoder and k-Nearest Neighbor untuk mendeteksi halangan yang ada di jalan menggunakan input citra digital. Penelitian lainnya dilakukan oleh Fitri Utamingrum dalam mendeteksi tangga turun. Penelitian tersebut menggunakan ekstraksi fitur *gray level co-occurrence matrix* untuk mengenali ciri tangga turunan[7].

Penelitian ini bertujuan untuk mencari kernel pada metode *Support Vector Machine* yang memberikan nilai akurasi, presisi, *recall* dan *f1-score* paling tinggi dalam melakukan deteksi tangga turunan dan lantai. Nantinya, akan terdapat 2 objek yang diklasifikasikan. Kedua objek tersebut adalah lantai sebagai objek negatif dan objek tangga turun sebagai objek positif.

Metode *Gray Level Co-occurrence Matrix* digunakan untuk mendapatkan nilai-nilai penciri dari tangga turun dan lantai. Dari nilai-nilai penciri tersebut yang nantinya akan diklasifikasikan menggunakan beberapa kernel pada metode SVM.

## 2. METODE PENELITIAN

Penyelesaian permasalahan pada penelitian ini menggunakan beberapa tahapan proses yang dilakukan. Masukan yang digunakan berupa citra digital yang merepresentasikan lantai dan tangga turun. Dimana nantinya akan dilakukan deteksi tangga turunan berbasis pengolahan citra digital. Tahapan proses yang dilakukan dapat dilihat pada gambar 1.



**Gambar 1.** Tahapan proses yang dilakukan

### 2.1. Pre-processing

Pada penelitian ini menggunakan data berupa citra digital sebanyak 500 foto. Terdapat 250 foto lantai dan 250 foto tangga turun. Dimana nantinya 400 foto akan digunakan sebagai data latih dan 100 foto akan digunakan sebagai data uji ketika metode klasifikasi *Support Vector Machine* dijalankan. Foto yang digunakan berukuran 480x640 piksel. Pada tahap *preprocessing* akan dilakukan proses *cropping* atau mencari *region of interest* pada koordinat 40,400 hingga 440,550. Sehingga akan terbentuk citra dengan ukuran 400x150 piksel. Selanjutnya citra hasil *cropping* tersebut akan dilakukan konversi kedalam bentuk *grayscale* seperti pada rumus 1[8]. Gambar 2 menunjukkan proses dimana dilakukan *cropping* pada citra masukan.

$$Grayscale = \frac{red+green+blue}{3} \quad (1)$$

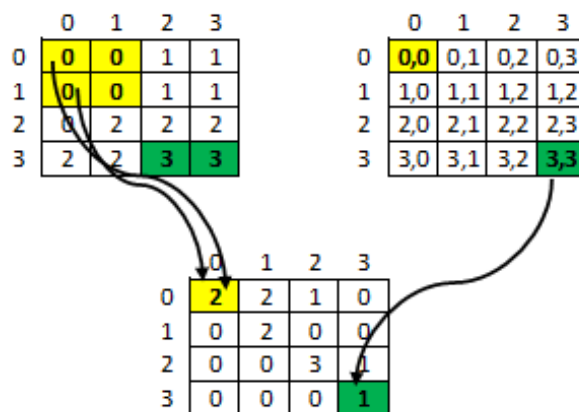


**Gambar 2.** Proses *cropping* pada citra

## 2.2. Gray Level Co-occurrence Matrix

Dalam membedakan antara lantai dan tangga turun dapat dilakukan dengan memperhatikan tekstur kedua objek tersebut. Tekstur yang dimiliki lantai cenderung datar dan rata, sedangkan tekstur tangga turun tidaklah datar dan cenderung terdapat patahan-patahan anak tangga. Oleh karena hal tersebut, penelitian ini menggunakan ekstraksi fitur yang dapat melakukan analisis terhadap tekstur. Metode yang kami usulkan yaitu metode *gray level co-occurrence matrix (GLCM)*. Dimana metode GLCM dapat melakukan ekstraksi fitur-fitur tekstur pada lantai atau tangga turunan[9][10]. Metode GLCM memiliki parameter sudut ( $\theta$ ) dan jarak ( $d$ ). Dimana kedua parameter tersebut menandakan perhitungan antar piksel yang dilakukan. Terdapat beberapa tahapan pada metode GLCM, tahapannya adalah sebagai berikut [11]:

1. Membentuk matrix GLCM berdasarkan parameter jarak dan sudut yang digunakan. Pada penelitian ini menggunakan sudut  $0^\circ$  dan jarak 1. Gambar 3 merupakan contoh proses pembentukan matrix GLCM.



**Gambar 3.** Matrix GLCM

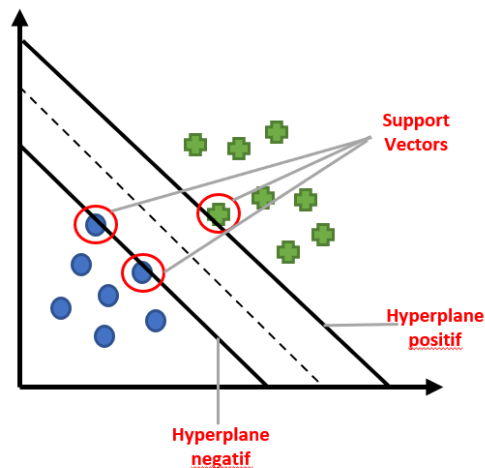
2. Tahap berikutnya yaitu membentuk matriks simetris GLCM dengan cara menjumlahkan matriks GLCM dengan matriks transposenya.
3. Tahap terakhir yaitu melakukan normalisasi terhadap matriks GLCM yang sudah simetris dengan cara membagi setiap elemen matriks dengan total keseluruhan elemennya.

Setelah mendapatkan matrix GLCM yang sudah dinormalisasi, selanjutnya yaitu melakukan ekstraksi fitur ciri. Penelitian ini menggunakan 7 jenis fitur untuk mengetahui penciri dari lantai dan tangga turunan. Fitur-fitur yang digunakan yaitu *contrast*, *energy*, *entropy*, *homogeneity*, *correlation*, *dissimilarity* dan *angular second moment*.

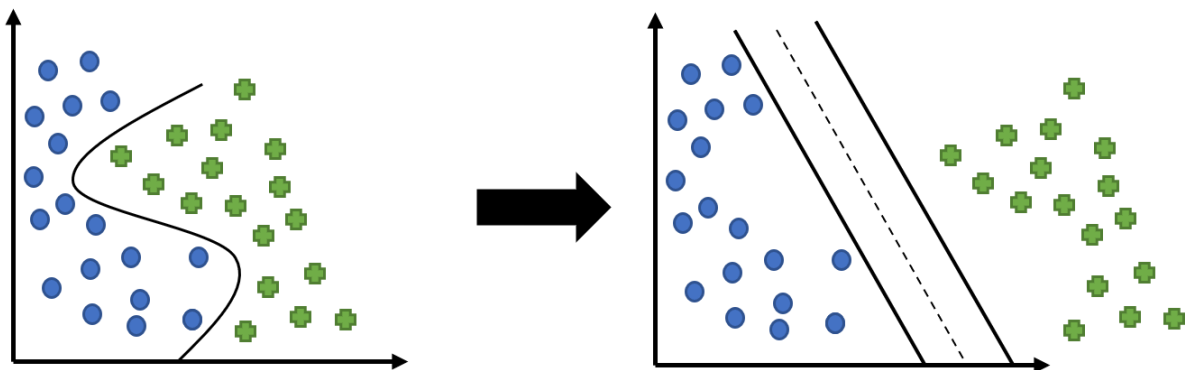
1. *Contrast* merupakan fitur yang menghitung derajat keabuan dari sebuah citra. Semakin dekat perbedaan nilai derajat keabuan, maka nilai *contrast* yang didapat akan semakin kecil. Begitu juga sebaliknya, apabila nilai derajat keabuannya jauh, maka nilai *contrast* semakin tinggi.
2. *Energy* merupakan nilai yang menggambarkan keteraturan penyebaran derajat suatu citra keabuan.
3. *Entropy* merupakan kebalikan dari *energy*. Dimana *entropy* menghitung keteracakan distribusi derajat keabuan suatu citra. Semakin acak distribusi derajat keabuannya, semakin tinggi nilai entropi yang dihasilkan
4. *Homogeneity* menghitung ukuran perulangan struktur yang bobot nilainya merupakan nilai invers dari *contrast*-nya
5. *Correlation* merupakan teknik untuk menghitung ketergantungan linier dari sebuah citra. Sehingga apabila semakin banyak pasangan piksel yang memiliki hubungan linier, maka nilai *correlation* akan semakin tinggi.
6. *Dissimilarity* melakukan pengukuran terhadap ketidakmiripan tekstur dari citra. Apabila tekstur pada citra berbentuk acak, maka nilai *dissimilarity* akan tinggi. Apabila tekstur berbentuk seragam, maka nilainya akan kecil.

### 2.3. Support Vector Machine

*Support Vector Machine* (SVM) merupakan salah satu metode klasifikasi bertipe *supervised learning*. Dimana SVM melakukan pelatihan terlebih dahulu menggunakan data latih yang disediakan untuk kemudian melakukan pengujian terhadap data uji[12]. SVM memiliki kelebihan dalam melakukan klasifikasi, dimana metode ini dapat menyelesaikan masalah klasifikasi dengan *linear* maupun *non-linear*[13]. Dalam metode SVM terdapat *hyperplane* yang berfungsi untuk memisahkan kelas-kelas yang ada. Oleh karena hal tersebut, maka diperlukan proses pelatihan terlebih dahulu untuk mendapatkan *hyperplane*. *Hyperplane* yang ditemukan diantara 2 kelas atau berada ditengah-tengah kelas seperti yang ditunjukkan pada Gambar 4, artinya jarak antara *hyperplane* dengan objek-objek data berbeda dengan kelas yang berdekatan (terluar) yang diberi tanda bulat kosong dan positif[14].



**Gambar 4.** Hyperplane yang memisahkan dua kelas positif (+1) dan negatif(-1)



**Gambar 5.** Fungsi kernel merubah sebaran data *non-linear* menjadi *linear*

Terdapat kasus yang *non-linear*, dimana diperlukan fungsi kusus untuk mencari *hyperplane*. Seperti gambar 5, dimana sebaran data tidak dapat diselesaikan dengan cara *linear*. Oleh karena hal tersebut diperlukan kernel untuk merubah sebaran data menjadi bentuk *linear*. Terdapat 4 kernel yang dapat digunakan, yaitu *linear*, *polynomial*, *gaussian* dan *sigmoid*. Rumus 2, 3, 4 dan 5 merupakan cara untuk merubah data yang *non-linear* menggunakan beberapa kernel[15].

$$\text{Linear} \quad : \quad K(x, y) = x \cdot y \quad (2)$$

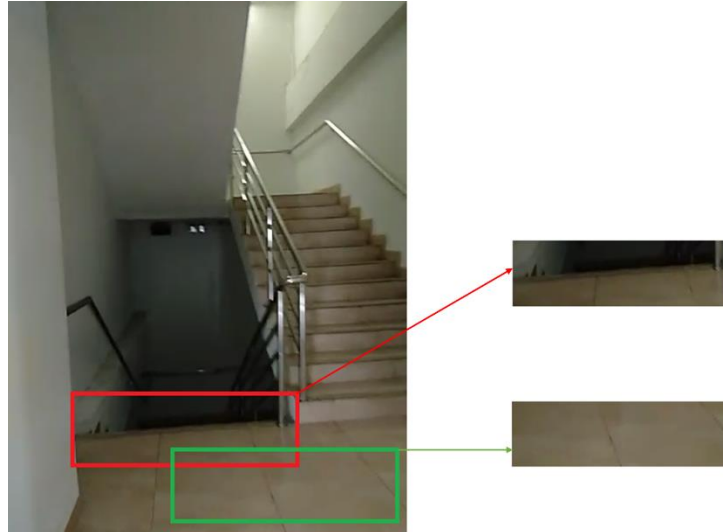
$$\text{Polynomial} \quad : \quad K(x, y) = (1 + x \cdot y)^d \quad (3)$$

$$\text{Gaussian} \quad : \quad K(x, y) = \exp\left(-\frac{|x-y|^2}{2\sigma^2}\right) \quad (4)$$

$$\text{Sigmoid} \quad : \quad K(x, y) = \tanh(ax \cdot y + \beta) \quad (5)$$

Setiap kernel memiliki cara yang berbeda-beda dalam mengolah sebaran data. Hal ini dapat berpengaruh terhadap akurasi yang didapat saat proses pengujian nantinya. Oleh karena itu, penelitian ini akan menganalisis terhadap kernel *linear*, *polynomial*, *gaussian* dan *sigmoid* untuk mencari kernel yang memberikan akurasi yang palik tinggi dalam mengklasifikasikan tangga turun dan lantai.

### 3. HASIL DAN PEMBAHASAN



**Gambar 6.** Sampel dataset

Penelitian ini menggunakan sebanyak 500 dataset. Dimana terdapat 250 gambar lantai dan 250 gambar tangga turun. Nantinya akan ada sebanyak 400 data digunakan sebagai data latih. Sedangkan data uji memiliki 50 gambar lantai dan 50 gambar tangga turun. Seluruh data gambar lantai dan tangga turun didapatkan dari 10 gedung yang berbeda-beda. Dimana setiap gedung memiliki karakteristik lantai dan tangga yang berbeda-beda. Gambar 6 merupakan contoh gambar yang digunakan dalam pelatihan dan pengujian. Dimana diambil ROI yang berwarna merah menjadi petanda kelas tangga turun. Sedangkan ROI hijau merupakan citra lantai. Penelitian ini melakukan pengujian untuk mengetahui 4 kernel SVM (*linear, polynomial, gaussian* dan *sigmoid*) yang memberikan akurasi, presisi, recall dan f1-score terbaik dalam mengklasifikasikan lantai dan tangga turun. Dalam mencari nilai akurasi rumus 6, mencari nilai presisi rumus 7, mencari nilai rumus 8, dan mencari nilai f1-score menggunakan rumus 9[16].

$$Akurasi = \frac{TN+TP}{TP+FP+TN+FN} \quad (6)$$

$$Presisi = \frac{TP}{TP+FP} \quad (7)$$

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

$$f1 - score = 2 * \frac{Precision \times Recall}{Precision+Recall} \quad (9)$$

Terdapat 2 kelas yang digunakan pada penelitian ini, yaitu kelas lantai dan kelas tangga turun. Dimana kelas lantai merupakan objek negatif dan kelas tangga turun merupakan kelas positif. Hal ini dikarenakan sistem ini dibuat untuk mendeteksi halangan tangga turunan yang diperuntukkan untuk orang buta. Simbol *TP* merupakan *true positif*, yang artinya gambar yang diuji merupakan gambar tangga turun dan terklasifikasikan benar sebagai tangga turun. Simbol *TN* merupakan *true negatif*, yang artinya gambar yang diuji merupakan lantai dan terklasifikasikan benar sebagai lantai. Simbol *FP* merupakan *false positif*, yang artinya gambar yang diuji merupakan gambar tangga turun dan terklasifikasikan salah (lantai). Simbol *FN* merupakan *false negatif*, yang artinya gambar yang diuji merupakan gambar lantai dan terklasifikasikan salah (tangga turun).

### 3.1. Hasil Pengujian

Pengujian dilakukan dengan menggunakan 2 parameter GLCM untuk mendapatkan 7 fitur (*contrast, energy, entropy, dissimilarity, correlation, homogeneity* dan *angular second moment*). Parameter jarak = 1 dan sudut = 0°. Selanjutnya yaitu melakukan klasifikasi dengan 4 kernel SVM (*linear, polynomial, gaussian* dan *sigmoid*). Tabel 1 menunjukkan hasil klasifikasi yang didapat dengan menggunakan 100 data uji. Dimana hasil akurasi setiap kernel cukup signifikan berbeda. Kernel *polynomial* mendapatkan akurasi terendah dengan nilai 78%. Kernel *sigmoid* mendapatkan akurasi 84%. Kernel *gaussian* mendapatkan akurasi 85%. Kernel *linear* mendapatkan akurasi yang paling tinggi yaitu 89%. Pengujian selanjutnya yaitu pengujian presisi menggunakan 4 kernel SVM. Dari hasil pengujian yang didapatkan dimana rata-rata presisi terbaik adalah kernel *linear* dengan rata-rata nilai presisi 91%. Pada tabel 1 juga menunjukkan bahwa setiap kernel dapat melakukan perhitungan presisi kelas tangga turun dengan benar terhadap data uji yang diberikan. Pada pengujian *recall*, rata-rata terbaik yaitu saat menggunakan kernel *linear*. Nilai rata-rata *recall* yang didapat yaitu 89%. Pada pengujian *recall*, keseluruhan kelas lantai mendapatkan nilai 100%. Selanjutnya yaitu pengujian *f1-score*, dimana kernel *linear* dapat menghasilkan nilai tertinggi yaitu 89%. Kernel *linear* dapat menghasilkan akurasi, presisi, *recall*, *f1-score* tertinggi dikarenakan sebaran data yang digunakan berbentuk *linear* seperti yang ditunjukkan pada gambar 4. Sehingga tidak perlu kernel yang mengubah jarak sebaran antar datanya. Garis *hyperplane* dapat langsung dibuat berdasarkan sebaran data yang sudah *linear*. Sedangkan kernel *polynomial* mendapatkan akurasi terendah. Hal ini disebabkan kernel *polynomial* lebih cocok untuk dataset yang sebaran nilai fiturnya tidak linier.

Tabel 1. Hasil Pengujian

Kernel	Akurasi	Presisi		Recall		f1-score	
		Lantai	Tangga Turun	Lantai	Tangga Turun	Lantai	Tangga Turun
Linear	89%	82%	100%	100%	78%	90%	88%
Polynomial	78%	69%	100%	100%	56%	82%	72%
Gaussian	85%	77%	100%	100%	70%	87%	82%
Sigmoid	84%	76%	100%	100%	62%	86%	81%

### 4. KESIMPULAN

Penelitian ini bertujuan untuk mencari kernel SVM yang memberikan akurasi yang paling optimum. Dimana terdapat 4 kernel yang digunakan, yaitu kernel *linear, polynomial, gaussian* dan *sigmoid*. Dari hasil pengujian yang telah dilakukan, didapatkan kernel *linear* memberikan akurasi yang paling tinggi, yaitu 89%. Sedangkan kernel *polynomial* mendapatkan akurasi yang paling rendah, yaitu 78%. Hasil ekstraksi fitur GLCM menunjukkan bahwa sebaran data antar 2 kelas yaitu lantai dan tangga turun sebarannya linier. Sehingga kernel *linear* cocok untuk melakukan klasifikasi tangga turun dan lantai. Dari hasil pengujian yang didapat, nantinya sistem deteksi tangga turun untuk orang buta akan dibuat dengan menggunakan klasifikasi SVM yang memakai kernel *linear*.

Penelitian ini melakukan komparasi 4 kernel yang tersedia pada metode klasifikasi SVM. Fitur-fitur yang didapatkan dari ekstraksi fitur GLCM belumlah dilakukan normalisasi. Penelitian selanjutnya diharapkan dilakukan normalisasi terhadap nilai-nilai fitur serta



dilakukan seleksi fitur. Hal ini diharapkan dapat menaikkan nilai ketepatan saat melakukan klasifikasi.

## REFERENSI

- [1] E. Retnoningsih and R. Pramudita, "Mengenal Machine Learning Dengan Teknik Supervised Dan Unsupervised Learning Menggunakan Python," *Bina Insa. Ict J.*, vol. 7, no. 2, p. 156, 2020, doi: 10.51211/biict.v7i2.1422.
- [2] H. Abijono, P. Santoso, and N. L. Anggreini, "Algoritma Supervised Learning Dan Unsupervised Learning Dalam Pengolahan Data," *J. Teknol. Terap. G-Tech*, vol. 4, no. 2, pp. 315–318, 2021, doi: 10.33379/gtech.v4i2.635.
- [3] "Kinerja Algoritma Support Vector Machine Berdasarkan Ekstraksi Sinyal Suara Dengan Mel-Frequency Cepstrum Coefficients Pada Pelafadzan Ayat Al-Quran," vol. 4, no. 2, pp. 280–284, 2019.
- [4] A. Perdana and M. T. Furqon, "Penerapan Algoritma Support Vector Machine ( SVM ) Pada Pengklasifikasian Penyakit Kejiwaan Skizofrenia ( Studi Kasus : RSJ . Radjiman Wediodiningrat , Lawang )," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 2, no. 9, pp. 3162–3167, 2018.
- [5] C. Savas and F. Dovic, "The impact of different kernel functions on the performance of scintillation detection based on support vector machines," *Sensors (Switzerland)*, vol. 19, no. 23, pp. 1–16, 2019, doi: 10.3390/s19235219.
- [6] A. Dairi, F. Harrou, Y. Sun, and M. Senouci, "Obstacle Detection for Intelligent Transportation Systems Using Deep Stacked Autoencoder and k-Nearest Neighbor Scheme," *IEEE Sens. J.*, vol. 18, no. 12, pp. 5122–5132, 2018, doi: 10.1109/JSEN.2018.2831082.
- [7] A. Wali, S. Bahari, F. Utaminingrum, and A. S. Budi, "K- Value Effect Based on Combination GLCM Angle and KNN for Detecting Smart Wheelchair," vol. 5, no. 1, pp. 23–31, 2020.
- [8] K. Kumar, R. K. Mishra, and D. Nandan, "Efficient Hardware of RGB to Gray Conversion Realized on FPGA and ASIC," *Procedia Comput. Sci.*, vol. 171, no. 2019, pp. 2008–2015, 2020, doi: 10.1016/j.procs.2020.04.215.
- [9] F. Utaminingrum, A. W. Satria Bahari Johan, I. K. Somawirata, Risnandar, and A. Septiarini, "Descending stairs and floors classification as control reference in autonomous smart wheelchair," *J. King Saud Univ. - Comput. Inf. Sci.*, no. xxxx, 2021, doi: 10.1016/j.jksuci.2021.07.025.
- [10] T. Ahmed, T. Rahman, B. B. Roy, and J. Uddin, "Drone Detection by Neural Network Using GLCM and SURF Features," *J. Inf. Syst. Telecommun.*, vol. 9, no. 33, pp. 15–23, 2021, doi: 10.52547/jist.9.33.15.
- [11] R. Widodo, A. W. Widodo, and A. Supriyanto, "Pemanfaatan Ciri Gray Level Co-Occurrence Matrix (GLCM) Citra Buah Jeruk Keprok (Citrus reticulata Blanco) untuk Klasifikasi Mutu," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 11, pp. 5769–5776, 2018, [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/3420>.
- [12] F. Yuko Pratama, A. FadliArie Hendra, F. Maulana, and A. Yusuf, "Sistem Pengawas Kapal Penangkap Ikan Berbasis AI Menggunakan Drone," *Lomba Karya Tulis Ilm.*, vol. 1, no. 1, pp. 77–83, 2020, [Online].
- [13] L. A. Demidova, I. A. Klyueva, and A. N. Pylkin, "Hybrid approach to improving the results

- of the SVM classification using the random forest algorithm,” *Procedia Comput. Sci.*, vol. 150, pp. 455–461, 2019, doi: 10.1016/j.procs.2019.02.077.
- [14] a adhitiawarman, d hartanto, and ..., “the Implementation of Naïve Bayes and Support Vector Machine (3. Svm) Algorithm, in Determining Achieving Students in Smp Negeri 8 ...,” *Jitk (Jurnal Ilmu ...)*, vol. 7, no. 1, pp. 1–6, 2021, doi: 10.33480/jitk.v7i1.2001.THE.
- [15] D. Kancherla, J. D. Bodapati, and N. Veeranjanyulu, “Effect of different kernels on the performance of an SVM based classification,” *Int. J. Recent Technol. Eng.*, vol. 7, no. 5, pp. 1–6, 2019.
- [16] R. E. Putra, H. Tjandrasa, N. Suciati, and A. Y. Wicaksono, “Non-Proliferative Diabetic Retinopathy Classification Based on Hard Exudates Using Combination of FRCNN, Morphology, and ANFIS,” *Proceeding - 2020 3rd Int. Conf. Vocat. Educ. Electr. Eng. Strength. Framew. Soc. 5.0 through Innov. Educ. Electr. Eng. Informatics Eng. ICVEE 2020*, 2020, doi: 10.1109/ICVEE50212.2020.9243185.