

PENGENALAN JALAN BERLUBANG BERBASIS *VISION* MENGGUNAKAN *PYRAMID HISTOGRAM OF ORIENTED GRADIENTS*

Ahmad Habib Fitriansyah¹, Ema Rachmawati^{*2}, Risnandar³

^{1,2,3}Universitas Telkom, Bandung, ³Badan Riset dan Inovasi Nasional, Bandung
Email: ¹ahmadhabib@student.telkomuniversity.ac.id, ²emarachmawati@telkomuniversity.ac.id,
³risnandartelyu@telkomuniversity.ac.id
^{*}Penulis Korespondensi

(Naskah masuk: 19 Desember 2022, diterima untuk diterbitkan: 21 Juni 2023)

Abstrak

Lubang, sejenis kerusakan jalan, dapat merusak kendaraan dan berdampak negatif pada keamanan mengemudi dari pengemudi. Bahkan, dalam kasus yang parah dapat menyebabkan kecelakaan lalu lintas. Manajemen jalan berlubang yang efisien dan preventif di lingkungan jalan yang kompleks memainkan peran penting dalam mengamankan keselamatan pengemudi. Hal ini juga diharapkan dapat memberikan kontribusi terhadap pencegahan kecelakaan lalu lintas dan kelancaran arus lalu lintas. Di masa lalu, deteksi lubang terutama dilakukan melalui inspeksi visual oleh ahli manusia. Baru-baru ini, metode deteksi lubang otomatis menerapkan berbagai teknologi yang menyatukan teknologi dasar seperti sensor dan pemrosesan sinyal. Pada artikel ini, metode berbasis pengolahan citra dan pembelajaran mesin diaplikasikan untuk mengenali lubang di jalan. Penelitian ini menghasilkan model dari bentuk lubang dengan memanfaatkan ciri bentuk yang diekstraksi dari *Pyramid Histogram of Oriented Gradients* (PHOG). Untuk metode klasifikasi, peneliti menggunakan *Support Vector Machine* (SVM) dengan hasil terbaik diperoleh pada penggunaan kernel polynomial. Sistem pengenalan jalan berlubang yang diusulkan mampu menunjukkan hasil performa yang sangat baik, yaitu akurasi sebesar 94,45%, *precision* sebesar 96,13% *recall* sebesar 95,77%, dan *F1-score* sebesar 95,95%.

Kata kunci: pengenalan jalan berlubang, PHOG, SVM

VISION BASED POTHOLE RECOGNITION USING PYRAMID HISTOGRAM OF ORIENTED GRADIENTS

Abstract

Potholes on roads can damage vehicles and endanger drivers, potentially leading to accidents. Preventative management of potholes is crucial for driver safety and efficient traffic flow. Traditional methods of pothole detection relied on visual inspection, but automatic methods have been developed using sensors and signal processing. This article presents a new approach using image processing and machine learning to identify potholes on roads. The proposed system uses shape features extracted from Pyramid Histogram of Oriented Gradients (PHOG) and a Support Vector Machine (SVM) with polynomial kernels for classification. The system achieves high accuracy, precision, recall, and F1-Score, with an accuracy of 94.45%, precision of 96.130%, recall of 95.77%, and F1-Score of 95.950%.

Keywords: potholes recognition, PHOG, SVM

1. PENDAHULUAN

Pada zaman sekarang ini, hampir seluruh keluarga memiliki kendaraan seperti mobil untuk memudahkan dalam perjalanan. Sehingga sangat dibutuhkannya jalan yang bagus untuk memberikan kenyamanan saat berkendara ataupun dalam perjalanan. Jalan juga merupakan infrastruktur yang berguna untuk menghubungkan antara kota-kota yang ada di seluruh dunia. Perjalanan darat juga

menjadi cara alternatif untuk berpergian ke luar kota selain kereta dan pesawat terbang.

Akan tetapi beberapa lokasi masih terdapat jalanan yang berlubang dan dapat meningkatkan angka kecelakaan di jalan. Menurut Badan Pusat Statistik (BPS), pada tahun 2017 terdapat 98.419 kali kecelakaan yang terjadi di jalan (Katadata, 2017). Dan salah satu penyebab tingginya angka kecelakaan adalah bentuk jalanan yang bergelombang ataupun

berlubang (Saputra, 2017). Selain itu, di India pada tahun 2016, 6.424 kecelakaan terjadi di jalan berlubang dan 2.324 orang meninggal dunia. Pada tahun 2015 terjadi 10.876 kejadian kecelakaan karena berlubang, 3.416 orang meninggal dunia karena hal tersebut dan pada tahun 2014 sebanyak 11.106 kecelakaan terjadi akibat berlubang dan 3.039 orang meninggal dunia (IANS, 2019). Sehingga dibutuhkan sebuah sistem yang dapat mengenali jalan berlubang atau membedakan antara lubang dengan bayangan pohon sehingga nanti dapat dijadikan rujukan kuat dalam sistem pendeteksi jalan berlubang.

Penelitian terkait pengenalan kerusakan jalan berbasis *vision* biasanya dilakukan dengan mengekstrak fitur dari objek terdefinisi pada citra atau mendeteksi keberadaan jenis kerusakan jalan pada citra. Sonja dkk melakukan penelitian pengenalan kerusakan jalan dengan membandingkan fitur *Histogram of Oriented Gradients* (HOG) dengan *Local Binary Patterns* (LBP), dan memanfaatkan SVM sebagai *classifier* (Sonja Nienaber, Booysen & Kroon, 2016). Skor F1 tertinggi yang berhasil didapat adalah 0,373. Selain itu, Pramestya dkk melakukan penelitian tentang klasifikasi kerusakan jalan menggunakan *Gray Level Co-Occurrence Matrix* (GLCM) dan *Radial Basis Function* (RBF). Dataset yang digunakan diambil di Google dengan kata kunci 'pothole', 'crack', dan 'road defect'. Dengan metode yang mereka usulkan, mereka berhasil mendapat rata-rata akurasi 93%, presisi 93%, dan recall 100% (Pramestya dkk., 2018). Penelitian tersebut hanya menggunakan 50 citra untuk *training* dan 15 citra untuk validasi. Sementara itu, Hoang dkk mengusulkan sistem deteksi pelepasan aspal di jalan berdasarkan analisis ciri tekstur (Hoang, 2019). Ciri tekstur diekstrak menggunakan GLCM untuk selanjutnya diklasifikasi menggunakan *Stochastic Gradient Descent Logistic Regression* (SGD-LR). Akurasi klasifikasi yang didapatkan dari eksperimen mencapai 88%. Baek & Chung membangun sistem klasifikasi lubang jalan menggunakan deteksi tepi (Baek & Chung, 2020), dengan nilai *area under curve* (AUC) sebesar 0,9 dan MSE pada rentang 0,2 – 0,44. Citra input pada eksperimen diubah ke dalam *grayscale* dan karakteristik dari lubang diekstrak dengan cara mendeteksi kontur dari lubang menggunakan metode deteksi tepi.

Di sisi lain, Wang dkk melakukan penelitian pengenalan aktivitas manusia (Wang dkk., 2011) menggunakan *Pyramid Histogram of Oriented Gradients* (PHOG). Mereka mendapatkan akurasi pengenalan tertinggi sebesar 94,4%. Shokrani dkk. melakukan penelitian tentang pengenalan emosi wajah menggunakan PHOG dengan memanfaatkan 3 (tiga) arah kepala (Shokrani, Moallem & Habibi, 2014). Mereka membandingkan kinerja dari PHOG

dan *Gabor Filter*. Mereka mendapatkan akurasi tertinggi dengan menggunakan PHOG dan KNN, yaitu sebesar 98,38%.

Berdasarkan penelitian yang disebutkan sebelumnya, bahwa metode PHOG mampu mengenali ciri bentuk dengan sangat baik, yang terbukti dari perolehan kinerja yang sangat baik untuk melakukan pengenalan emosi wajah dan gerakan manusia. Peneliti memutuskan untuk menggunakan metode PHOG pada sistem pengenalan kerusakan jalan untuk mengetahui kehandalan PHOG dalam mengenali jalan berlubang.

Susunan makalah adalah sebagai berikut. Deskripsi sistem pengenalan jalan berlubang yang diusulkan terdapat pada bagian 2. Eksperimen dan analisis pada bagian 3 dan ditutup dengan kesimpulan pada bagian 4.

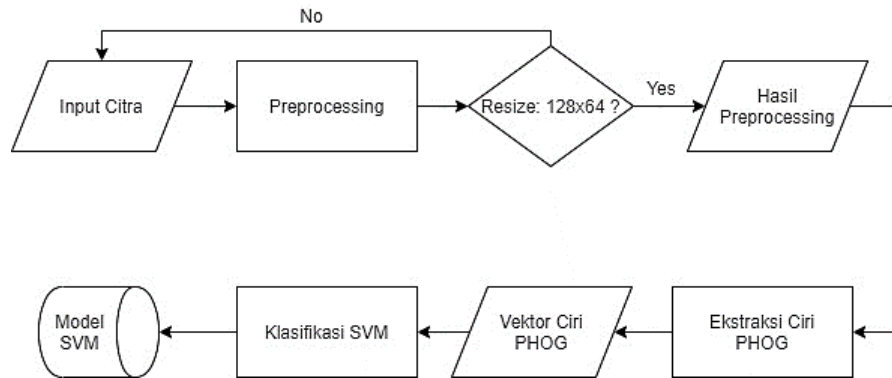
2. SISTEM PENGENALAN JALAN BERLUBANG

Pada sistem pengenalan kerusakan jalan ini dibagi menjadi dua tahap, yaitu tahap pelatihan dan tahap pengujian. Pada tahap pelatihan (Gambar 1), citra di-*crop* sesuai dengan titik koordinat yang diberikan di dalam dataset, untuk selanjutnya dilakukan *resize* citra ke ukuran 128x64 piksel. Selanjutnya citra dilakukan pengecekan apakah sudah berukuran 128x64 piksel atau belum. Jika belum maka kembali ke proses input citra dan jika sudah sesuai maka lanjut ke tahap pengambilan ekstraksi fitur dengan PHOG, untuk kemudian diklasifikasikan menggunakan SVM.

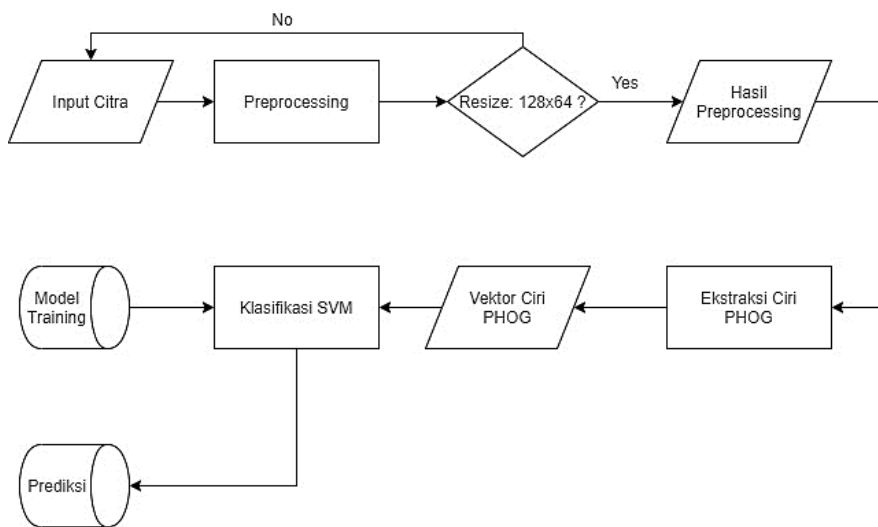
Tahap pengujian pada penelitian ini merupakan tahap terakhir yang dilakukan untuk melakukan pengenalan kerusakan jalan. Pada tahap ini disimpulkan apakah citra uji tersebut mengandung lubang atau tidak. Tahapan tersebut digambarkan pada blok diagram di Gambar 2.

2.1. Prapemrosesan

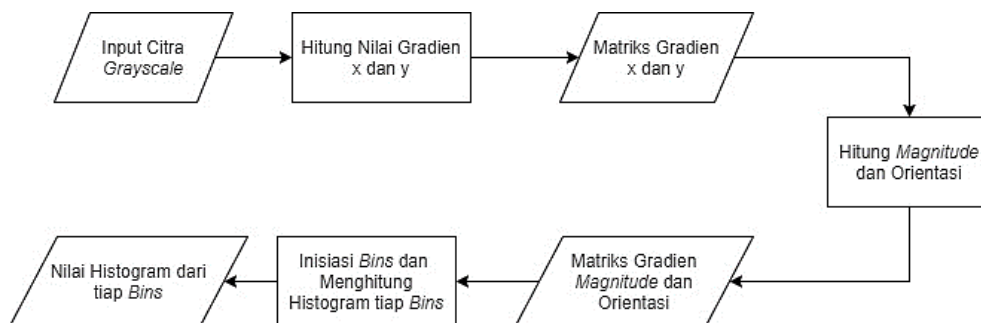
Pada tahap ini dilakukan *cropping* citra berdasarkan titik koordinat yang diberikan pemilik dataset. Citra juga diubah ke dalam *grayscale*, dengan tujuan untuk mempercepat proses komputasi karena PHOG tidak membutuhkan warna citra untuk mendapatkan fitur dari citra tersebut. Selanjutnya, citra tersebut dilakukan *resizing*, agar proses komputasi lebih cepat karena jumlah piksel yg diekstrak tidak besar.



Gambar 1. Blok diagram untuk pelatihan



Gambar 2. Blok diagram untuk pengujian

Gambar 3. Blok diagram *Histogram of Oriented Gradients*

2.2. Ekstraksi Ciri

Pada tahap ekstraksi ciri ini metode yang digunakan adalah PHOG. PHOG adalah metode ekstraksi ciri bentuk yang ide dasarnya dari metode HOG (Dalal & Triggs, 2005). Alur proses yang terjadi pada HOG dapat dilihat pada Gambar 3.

Langkah-langkah untuk mendapat fitur dari HOG adalah menghitung nilai gradien horizontal dan vertikal dengan menggunakan *sobel mask* 3 x 3.

Untuk menghitung *magnitude* menggunakan persamaan (1).

$$|G| = \sqrt{G_x^2 + G_y^2} \quad (1)$$

G_x adalah gradien horizontal dan G_y adalah gradien vertikal. Selanjutnya adalah menghitung orientasi dengan menggunakan persamaan (2).

$$\theta = \arctan\left(\frac{G_y}{G_x}\right) \quad (2)$$

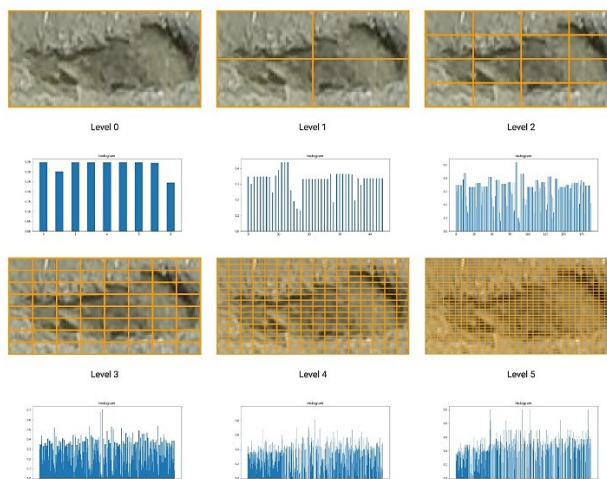
Selanjutnya, dilakukan kuantisasi orientasi *gradient* dalam K bins pada setiap *cell* pada *range* orientasi tertentu (Wahyuni, Alinra & Setiawan, 2018). Setiap *bins* menunjukkan jumlah tepi yang memiliki nilai orientasi berdasarkan *range* orientasi yang ditentukan. Untuk penelitian ini, menggunakan 9 *bins* pada *range* orientasi 0-180. Selanjutnya, dilakukan normalisasi *L2-Norm* dengan menggunakan persamaan (3).

$$f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}} \quad (3)$$

v adalah fitur *histogram* yang dihasilkan dan e adalah bilangan natural. Setelah mendapatkan semua fitur *histogram*, lalu digabungkan fitur setiap level. Panjang fitur dari PHOG dihasilkan dengan menggunakan persamaan (4).

$$d = K \sum_{l=0}^L 4^l \quad (4)$$

d adalah jumlah dimensi dari fitur, K adalah jumlah *bins* dari orientasi gradien, dan L adalah jumlah level dari *pyramid* PHOG (Wang dkk., 2011). Sebagai contoh jika diketahui $K = 9$ dan $L = 5$, maka fitur yang didapat sebesar 3096 dimensi.



Gambar 4. Perbedaan setiap level di PHOG

Ide dasar dari PHOG adalah metode HOG yang digabungkan dari level 0 sampai level yang ditentukan. Untuk mendapatkan ciri dari PHOG, skema pada Gambar 3 atas diulang sebanyak jumlah level yang diinginkan sesuai dengan pembagian *cell*. Jumlah *bins* yang digunakan adalah 9. Untuk mendapatkan ciri yang lebih spesifik, sistem ini menggunakan level 5 dari PHOG. Gambar 4 menunjukkan visualisasi perbedaan level dalam PHOG.

Tabel 1. Ciri PHOG di setiap level

Level PHOG	Penggabungan Fitur	Jumlah Fitur
Level 0	HOG (1,1)	9
Level 1	HOG (1,1), HOG (2,2)	9 + 36 = 45
Level 2	HOG (1,1), HOG (2,2), HOG (4,4)	45 + 144 = 189
Level 3	HOG (1,1), HOG (2,2), HOG (4,4), HOG (8,8)	189 + 586 = 765
Level 4	HOG (1,1), HOG (2,2), HOG (4,4), HOG (8,8), HOG (16,16)	765 + 2.340 = 3.096
Level 5	HOG (1,1), HOG (2,2), HOG (4,4), HOG (8,8), HOG (16,16), HOG (32,32)	3.096 + 9.186 = 12.285

Level 5 dari PHOG tersebut menghasilkan 12.285 ciri dari setiap citra. Jumlah ini didapat dari penggabungan ciri dari level 0 – level 5. Perbedaan jumlah ciri yang dihasilkan tiap level, dijelaskan dalam Tabel 1.

Sementara itu, untuk keperluan mencari kemiripan antar vector ciri, digunakan *Euclidean Distance*. Rumus untuk *Euclidean Distance* terdapat pada persamaan (5), dengan p dan q adalah vektor ciri yang akan dibandingkan jaraknya. *Euclidean Distance* didapat dengan membandingkan jarak vektor fitur rata-rata citra yang terprediksi benar dengan citra yang salah prediksi.

$$Euclidean = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (5)$$

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

2.3. Metrik Evaluasi

Untuk mengukur kinerja dari sistem ini, peneliti menggunakan perhitungan evaluasi dari akurasi. Akurasi merupakan rasio prediksi benar (*true positive* (TP) dan *true negative* (TN)) dengan keseluruhan data uji yang diberikan. Perhitungan evaluasi menggunakan akurasi dapat dilihat pada persamaan (6).

3. EKSPERIMEN DAN ANALISIS

Pada bagian ini dijelaskan data yang digunakan pada eksperimen, skenario pada eksperimen, dan hasil beserta analisis.

3.1. Data

Pada penelitian ini, dataset yang digunakan adalah dataset dari penelitian deteksi jalan berlubang (Sonja Nienaber, Booysen & Kroon, 2016). Dataset tersebut diambil menggunakan Go-Pro Hero 3+ dengan resolusi 3680 x 2760 piksel dengan format .jpg. Di dalam dataset ini terdapat dua kelas, yaitu kelas positif dan negatif berjumlah 8.094 citra.

Karena dataset yang digunakan sebenarnya untuk permasalahan deteksi, peneliti akan melakukan *cropping* untuk citra latih dan uji positif (citra berlubang) sesuai dengan titik koordinat yang diberikan pemilik dataset. Dataset yang digunakan terdiri 3.074 citra latih positif, 6.193 citra latih negatif, 1325 citra uji positif, dan 604 citra uji negatif.

Untuk citra latih negatif kami melakukan *cropping* dengan koordinat yang sama, hal ini dilakukan karena citra latih negatif pada titik koordinat yang sama sudah mewakili berbagai macam kondisi jalan yang tidak memiliki lubang.

Pada tahap ini, yang pertama kali dilakukan adalah *cropping* citra. *Cropping* dilakukan sesuai dengan koordinat yang diberikan pemilik dataset. Setelah dilakukan *cropping* jumlah dataset menjadi bertambah, karena dalam 1 (satu) citra jalan dimungkinkan terdapat lebih dari 1 (satu) kondisi lubang. Dari 8.094 citra menjadi 11.196 citra dengan pembagian 3.074 data latih positif, 6.193 data latih negatif, 1.325 data uji positif, dan 604 data uji negatif. Pembagian tersebut berdasarkan pembagian dataset sebelumnya. Kelas positif diberi label 1 sedangkan kelas negatif diberi label 0. Contoh citra kelas positif dapat dilihat pada Gambar 5, sedangkan contoh citra kelas negatif dapat dilihat pada Gambar 6.

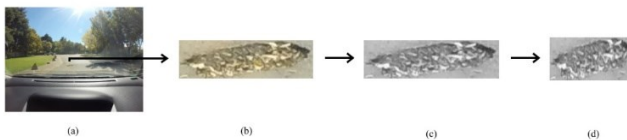


Gambar 5. Contoh citra kelas positif



Gambar 6. Contoh citra kelas negatif

Setelah dilakukan *cropping*, selanjutnya dari citra hasil *cropping* tadi dilakukan *resizing* ke ukuran 128 x 64 piksel. Hal ini dilakukan agar mempercepat komputasi. Ilustrasi tahap *preprocessing* dapat dilihat pada Gambar 7.



Gambar 7. *Preprocessing* (a) Citra Awal (b) Citra setelah di *crop* (c) Citra *Grayscale* (d) Citra setelah dilakukan *resizing*.

3.2. Hasil Eksperimen

Dalam penelitian ini, ada 4 (empat) skenario yang dilakukan, yaitu pengaturan ukuran blok citra, pengaturan level dari PHOG, pengaturan *image enhancement* dan pengaturan *hyperparameter* SVM.

1. Skenario 1: pengaturan ukuran blok citra

Pada tahap ini, peneliti menguji pengaruh proses *resizing* ke ukuran 128 x 64 piksel. Maka dari itu dilakukanlah pengujian dengan ukuran citra 32 x 16 piksel, 64 x 32 piksel, 128 x 64 piksel, dan 256 x 128 piksel. Dengan PHOG *level* 3 dan menggunakan metode SVM dengan kernel *linear* dengan $C = 1$, dan $\gamma = 1$, hasil pengujian dapat dilihat pada Tabel 2.

Dari hasil eksperimen pengaruh ukuran citra di Tabel 2 terlihat bahwa ukuran citra 128x64 piksel menunjukkan hasil performa yang lebih baik dibanding ukuran citra lainnya. Oleh karenanya, pada skenario eksperimen lainnya, ukuran citra 128x64 inilah yang digunakan.

Tabel 2. Hasil pengujian perbedaan ukuran blok citra. Nilai akurasi (Acc), *Precision* (P), *Recall* (R), dan F1 dinyatakan dalam persentase (%)

Ukuran blok	Acc	P	R	F1	Running Time (s)
32x16	70,08	73,70	87,77	80,12	119,78
64x32	86,31	89,67	90,49	90,08	120,63
128x64	91,03	92,92	94,11	93,51	119,83
256x128	87,97	87,25	96,60	91,69	138,88

2. Skenario 2: pengaturan level dari PHOG

Pada skenario ini, hal yang dilakukan adalah melakukan pengujian menggunakan metode PHOG dari level 0 sampai level 5. Hal ini dilakukan untuk mengetahui perbedaan performa sistem dari level 0 sampai level 5. Hasil dari skenario ini dapat dilihat pada Tabel 3.

Tabel 3. Hasil pengujian PHOG per *level*. Nilai akurasi (Acc), *Precision* (P), *Recall* (R), dan F1 dinyatakan dalam persentase (%)

Level	Acc	P	R	F1	Jumlah Fitur	Running Time (s)
0 (1,1)	31,67	70,58	0,90	1,7	9	76,16
1 (2,2)	53,86	90,96	36,45	52,04	45	79,12
2 (4,4)	82,68	93,88	80	86,38	189	89,91
3 (8,8)	91,03	92,92	94,11	93,51	765	119,93
4 (16,16)	91,49	92,15	95,77	93,93	3096	234,89
5 (32,32)	90,72	91,10	95,84	93,41	12285	682,27

Pada Tabel 3, dapat dilihat bahwa level 4 dari PHOG memiliki akurasi terbaik dengan 91,49%. Nilai *precision*, *recall*, dan F1 juga lebih baik dibanding level PHOG lainnya. Kernel yang digunakan pada SVM adalah kernel *linear* dengan $C = 1$, dan $\gamma = 1$.

3. Skenario 3: pengaturan *Image Enhancement*

Pada skenario ini, hal yang dilakukan peneliti adalah melakukan pengujian antara *sharpening* dan *smoothing*. Hal ini dilakukan agar peneliti mengetahui

pengaruh prapemrosesan citra terhadap kinerja sistem secara keseluruhan. Hasil dari skenario ini dapat dilihat pada Tabel 4.

Tabel 4. Pengaruh *sharpening* dan *smoothing*. Nilai akurasi (Acc), *Precision* (P), *Recall* (R), dan F1 dinyatakan dalam persentase (%)

	Acc	P	R	F1
<i>Smoothing</i>	93,10	93,69	96,45	95,05
<i>Sharpening</i>	90,40	91,24	95,16	93,16
Tanpa <i>Image Enhancement</i>	91,49	92,15	95,77	93,93

Dapat dilihat pada Tabel 4 bahwa akurasi terbaik pada proses *smoothing*. Hal ini terjadi karena pada proses *smoothing*, citra menjadi *soft* dan *noise* yang ada pada citra menjadi berkurang. Selain itu, nilai dari fitur menjadi semakin besar yang menyebabkan sistem dapat membedakan kelas jalan berlubang dengan mudah. Terdapat perbedaan akurasi hampir 2% jika dibandingkan dengan sistem tanpa *Image Enhancement*. Gambar 8 menunjukkan perbedaan *histogram* antara citra lubang dengan *smoothing*, citra lubang dengan *sharpening*, citra lubang tanpa *Image Enhancement*.

4. Skenario 4: pengaturan *hyperparameter* SVM

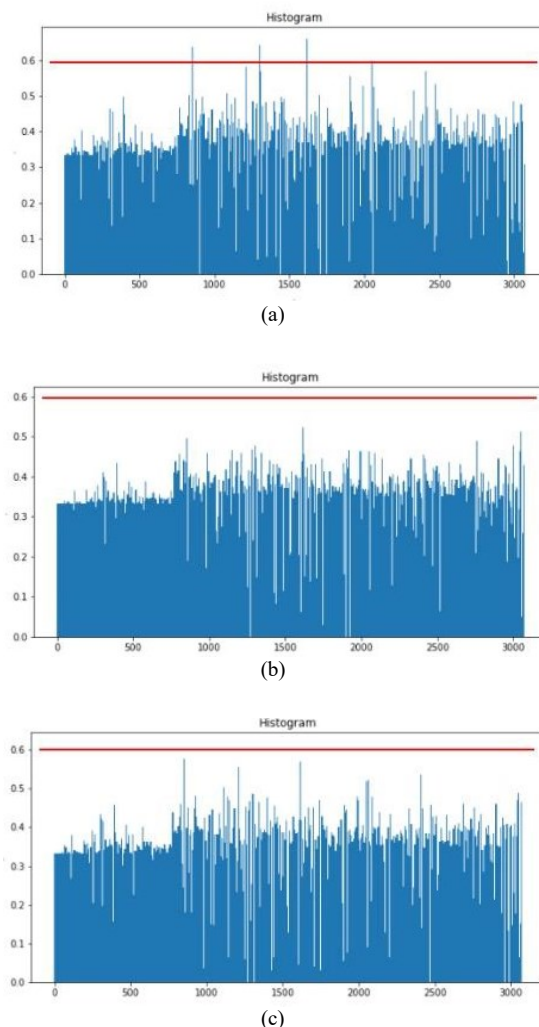
Untuk skenario ini, peneliti melakukan pengaturan *hyperparameter* untuk SVM. Hal ini dilakukan untuk mencari kernel yang terbaik. Untuk mendapatkan parameter terbaik SVM untuk penelitian ini, peneliti menggunakan bantuan *GridSearchCV*. Kernel yang diujicoba adalah *RBF*, *linear*, *polynomial*, dan *sigmoid*.

Hasil perbandingan kinerja sistem pada penggunaan kernel SVM dapat dilihat pada Tabel 5. Berdasarkan Tabel 5, dapat dilihat bahwa kernel *polynomial* mendapatkan akurasi tertinggi yaitu 94,45%. Dengan menggunakan kernel *polynomial*, $C = 10$, dan $\gamma = 0.001$, sistem ini mendapatkan akurasi tertingginya.

Selain itu, pada skenario ini juga dilakukan pengujian penggunaan berbagai kernel pada PHOG dengan HOG. Hal ini bertujuan untuk mengetahui apakah penggunaan PHOG sebagai vektor ciri pada sistem pengenalan jalan berlubang mampu memang mengungguli penggunaan HOG. Hasil dari perbandingan kinerja antara PHOG dengan HOG *Single Level* dapat dilihat pada Tabel 6.

Tabel 5. Hasil perbandingan kinerja sistem pada kernel berbeda. Nilai akurasi (Acc), *Precision* (P), *Recall* (R), dan F1 dinyatakan dalam persentase (%)

Kernel	Acc	P	R	F1
Linear	93,10	93,69	96,45	95,05
RBF	94,24	94,83	96,90	95,85
Polynomial	94,45	96,13	95,77	95,95
Sigmoid	93,77	94,99	96	95,49



Gambar 8. Perbedaan *histogram* antara citra lubang dengan (a) *smoothing*, (b) citra lubang dengan *sharpening*, (c) citra lubang tanpa *Image Enhancement*

Dari Tabel 6 dapat dilihat bahwa PHOG memiliki kinerja yang lebih baik dari HOG *Single Level*. Hal ini terjadi karena pada dasarnya PHOG memiliki ciri yang lebih kaya dan beragam, karena mengambil semua ciri dari *level 0 – level 4*, sedangkan HOG hanya mengambil ciri dari *level 4* saja.

Tabel 6. Hasil perbandingan kinerja PHOG dan HOG *Single Level* pada berbagai kernel SVM. (1) kernel linear, (2) kernel RBF, (3) kernel Polynomial, dan (4) kernel sigmoid. Semua nilai dinyatakan dalam persentase (%)

K	PHOG				HOG			
	Acc	P	R	F1	Acc	P	R	F1
1	93,10	93,69	96,45	95,05	91,55	92,10	95,92	93,97
2	94,24	94,83	96,90	95,85	93,41	94,17	96,37	95,26
3	94,45	96,13	95,77	95,95	92,79	96,32	93,05	94,66
4	93,77	94,99	96	95,49	92,74	93,92	95,62	94,76

4.2 Analisis Hasil Pengujian

Confusion matrix dari hasil pengujian terbaik, yaitu dengan citra berukuran 128 x 64 piksel dengan PHOG *level 5*, kernel RBF, $C = 10$, dan $\gamma = 0,001$, dapat dilihat pada Gambar 9. Dari hasil pengujian, kinerja terbaik didapatkan dengan nilai akurasi sebesar 94,45% dengan nilai *precision* sebesar 96,13% dan *recall* sebesar 95,77%. Detil kinerja sistem dengan hasil terbaik dapat dilihat pada Tabel 7.

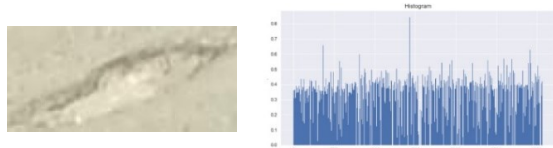


Gambar 9. *Confusion Matrix*

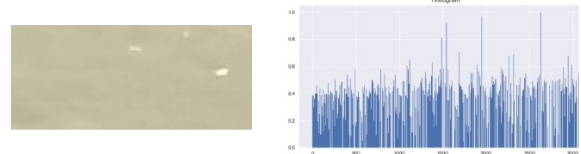
Tabel 7. Evaluasi kinerja sistem

	P	R	Jumlah Data
Bukan Lubang	0.91	0.92	604
Lubang	0.96	0.96	1325
Weighted Avg	0.96	0.95	1929

Pada Gambar 9 dapat dilihat bahwa terdapat 51 citra bukan lubang yang terprediksi sebagai lubang, dan terdapat 56 citra lubang yang terprediksi sebagai citra bukan lubang. Untuk mencari kemiripan antar ciri ini, digunakan *Euclidean Distance* (persamaan 5). Berdasarkan nilai *Euclidean Distance* yang didapat, fitur *histogram* dari citra lubang memiliki kemiripan yang lebih dekat ke citra bukan lubang dan sebaliknya. Hal ini menyebabkan salahnya prediksi dari sistem ini. Contoh citra yang salah diprediksi beserta hasil penghitungan skor *Euclidean Distance*-nya dapat dilihat pada Gambar 10 dan Gambar 11.



Gambar 10. Contoh kesalahan pengenalan jalan berlubang sebagai jalan normal. Jarak *Euclidean* dari vektor ciri ke kelas aktual adalah 0,0306, sedangkan jarak *Euclidean* ke kelas prediksi adalah 0,0048.



Citra uji

Histogram vector ciri

Gambar 11. Contoh kesalahan pengenalan jalan normal sebagai jalan berlubang. Jarak *Euclidean* dari vektor ciri ke kelas aktual adalah 0,0367, sedangkan jarak *Euclidean* ke kelas prediksi adalah 0,0012.

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, kinerja dari sistem yang dibangun mendapat akurasi terbaik yaitu 94,45%, *precision* sebesar 96,13%, *recall* sebesar 95,77%, dan *F1-Score* sebesar 95,95%. Hasil terbaik itu didapat menggunakan kernel *polynomial* dengan $\gamma = 0.001$ dan $C = 10$ dengan ukuran blok citra 128 x 64 piksel. Jika dilihat dari *recall* dan *precision*, maka kelas ditangani dengan sempurna oleh model yang dibangun menggunakan PHOG. Dengan ini, sistem yang dibangun dapat melakukan pengenalan terhadap jalan berlubang atau membedakan antara lubang dan bayangan pohon dengan cukup baik.

Saran untuk penelitian selanjutnya, dapat menggunakan jenis kerusakan jalan yang lebih banyak selain memperbanyak variasi dari citra berlubang. Selain itu, dapat juga menghitung *volume* dari lubang yang dikenali untuk membantu petugas perbaikan jalan dalam melakukan estimasi biaya perbaikan pada jalan tersebut.

DAFTAR PUSTAKA

- BAEK, J.W. & CHUNG, K., 2020. Pothole Classification Model Using Edge Detection in Road Image. *Applied Sciences* 2020, Vol. 10, Page 6662, [online] 10(19), p.6662. <https://doi.org/10.3390/AP10196662>.
- DALAL, N. & TRIGGS, B., 2005. Histograms of oriented gradients for human detection. In: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. pp.886–893 vol. 1. <https://doi.org/10.1109/CVPR.2005.177>.
- HOANG, N.D., 2019. Automatic detection of asphalt pavement raveling using image texture based feature extraction and stochastic gradient descent logistic regression. *Automation in Construction*, 105, p.102843.

- https://doi.org/10.1016/J.AUTCON.2019.102843.
- IANIS, 2019. *2,015 pedestrians lost their lives due to potholes in 2018: Govt | Business Standard News*. [online] Available at: <https://www.business-standard.com/article/current-affairs/2-015-pedestrians-lost-their-lives-due-to-potholes-in-2018-govt-119120200747_1.html> [Accessed 15 December 2022].
- KATADATA, 2017. *Sepanjang 2017 Terjadi 98 Ribu Kali Kecelakaan Lalu Lintas*. [online] Available at: <https://databoks.katadata.co.id/datapublish/2018/09/21/sepanjang-2017-terjadi-98-ribu-kali-kecelakaan-lalu-lintas> [Accessed 15 December 2022].
- PRAMESTYA, R.H., SULISTYANINGRUM, D.R., SETIYONO, B., MUKHLASH, I. & FIRDAUS, Z., 2018. Road defect classification using Gray Level Co-Occurrence Matrix (GLCM) and Radial Basis Function (RBF). *Proceedings of 2018 10th International Conference on Information Technology and Electrical Engineering: Smart Technology for Better Society, ICITEE 2018*, pp.285–289. https://doi.org/10.1109/ICITEED.2018.8534769.
- SAPUTRA, A.D., 2017. Studi Tingkat Kecelakaan Lalu Lintas Jalan di Indonesia Berdasarkan Data KNKT (Komite Nasional Keselamatan Transportasi) dari Tahun 2007-2016. *Warta Penelitian Perhubungan*, [online] 29(2). Available at: <https://garuda.kemdikbud.go.id/documents/detail/895039> [Accessed 15 December 2022].
- SHOKRANI, S., MOALLEM, P. & HABIBI, M., 2014. Facial emotion recognition method based on Pyramid Histogram of Oriented Gradient over three direction of head. *Proceedings of the 4th International Conference on Computer and Knowledge Engineering, ICCKE 2014*, pp.215–220. https://doi.org/10.1109/ICCKE.2014.6993346.
- SONJA NIENABER, B., BOOYSEN, M. & KROON, R., 2016. *Detecting potholes with monocular computer vision: A Performance evaluation of techniques*. [online] Available at: <https://scholar.sun.ac.za:443/handle/10019.1/98456> [Accessed 15 December 2022].
- WAHYUNI, E.S., ALINRA, R.R. & SETIAWAN, H., 2018. People counting for indoor monitoring. *3rd International Conference on Computing, Engineering, and Design, ICCED 2017*, 2018-March, pp.1–5. https://doi.org/10.1109/CED.2017.8308112.
- WANG, J., LIU, P., SHE, M.F.H., KOUZANI, A. & NAHAVANDI, S., 2011. Human action recognition based on pyramid histogram of oriented gradients. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, pp.2449–2454. https://doi.org/10.1109/ICSMC.2011.6084045.