

APLIKASI DETEKSI KERUSAKAN JALAN RAYA MENGGUNAKAN ALGORITMA K-NN (K - NEAREST NEIGHBOUR)***ROAD DETECTION APPLICATION USING K-NN ALGORITHM (K - NEAREST NEIGHBOUR)*****Aryo Nur Utomo, Nina Lestari**

Program Studi Sistem Informasi, Fakultas Sains dan Teknologi Informasi
 Institut Sains dan Teknologi Nasional
 Jl. Moh. Kahfi II, Bhumi Srengseng Indah, Jakarta Selatan 12640
 Telp. (021) 7874647, Fax. (021) 7866955
 aryo.nurutomo@istn.ac.id, ninalestari@gmail.com

ABSTRAK

Pertumbuhan teknologi yang semakin maju membuat banyak instansi terbantu seperti sistem deteksi cerdas pada suatu citra. Deteksi yang dimaksud ialah deteksi kerusakan atau deteksi lubang pada permukaan jalan raya. Untuk memenuhi hal tersebut, maka dibangun aplikasi dengan metode algoritma K-NN (K-Nearest Neighbour). Dalam mengenali objek yang diamati dilakukan proses segmentasi citra dengan menggunakan Grayscale. Cara kerja aplikasi deteksi kerusakan ini dilakukan dengan proses pengambilan gambar oleh user menggunakan kamera smartphone, kemudian data gambar diproses ke dalam aplikasi untuk dilakukan grayscale dan pengambilan bentuk citra rectangle sehingga dapat mendeteksi objek yang didukung oleh modul berbasis Computer Vision dari Python. Hasil dari penelitian ini supaya dapat memperoleh sistem deteksi yang lebih akurat dan dapat membedakan bentuk kerusakan.

Kata Kunci: Klasifikasi, Komputer Visi, Pembelajaran Mesin, Algoritma Klasifikasi, Citra, Python.

ABSTRACT

The development of increasingly advanced technology has helped many agencies such as intelligent detection systems in an image. The detection in question is the detection of damage or detection of holes on the surface of the highway. To fulfill this, an application using the K-NN (K-Nearest Neighbor) algorithm was built. In recognizing the object being observed, the image segmentation process is carried out using Grayscale. The way this damage detection application works is done by taking pictures by the user using a smartphone camera, then the image data is processed into the application for grayscale and taking the shape of the rectangle image so that it can detect objects that are supported by the Computer Vision based module from Python. The results of this study are in order to obtain a more accurate detection system and be able to distinguish the form of damage.

Keywords: Classification, Computer Vision, Machine Learning, Classification Algorithms, Image, Python.

1. PENDAHULUAN

Jalan merupakan salah satu prasarana transportasi yang menghubungkan antara satu tempat ke tempat lainnya. Jalan sebagai prasarana transportasi berpengaruh besar terhadap kemajuan dan perkembangan pada suatu daerah. Transportasi darat menggunakan jalan sebagai prasarana untuk mengangkut kebutuhan pokok masyarakat Indonesia seperti sandang dan pangan.

Pemeriksaan jalan selama ini masih menggunakan cara tradisional yaitu dengan pengamatan secara manual dengan indra penglihatan. Cara tradisional ini dirasa kurang efektif karena membutuhkan biaya yang lumayan besar, membutuhkan waktu yang lama, berbahaya karena intensitas kendaraan berlalu lalang yang tinggi, faktor subyektifitas dan faktor kelelahan.

Berdasarkan permasalahan tersebut maka diperlukan penelitian untuk menemukan metode baru guna menggantikan metode tradisional yang handal dan mempunyai tingkat akurasi tinggi. Klasifikasi

adalah salah satu metode yang sangat fundamental dalam memecahkan berbagai permasalahan.

Klasifikasi sendiri dapat diartikan sebagai metode untuk menyusun data secara sistematis menurut aturan-aturan yang telah ditetapkan sebelumnya. Sampai saat ini metode klasifikasi telah terbukti banyak membantu pekerjaan manusia khususnya dengan algoritma K- Nearest Neighbour (K-NN), misalnya untuk klasifikasi image, audio, dokumen, biologi medis, lampu lalu lintas, dll. Algoritma K-Nearest Neighbors (K-NN) adalah algoritma yang berfungsi untuk melakukan klasifikasi suatu data berdasarkan data pembelajaran (train data sets), yang diambil dari k tetangga terdekatnya (nearest neighbors). Dengan k merupakan banyaknya tetangga terdekat.

K-Nearest Neighbors melakukan klasifikasi dengan proyeksi data pembelajaran pada ruang berdimensi banyak. Ruang ini dibagi menjadi bagian-bagian yang merepresentasikan kriteria data pembelajaran. Setiap data pembelajaran

direpresentasikan menjadi titik-titik c pada ruang dimensi banyak. Algoritma K-Nearest Neighbor (K-NN) adalah salah satu metode yang menerapkan algoritma *supervised learning* dimana hasil dari sampel uji yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada K-NN. Ketepatan algoritma KNN ditentukan oleh ada dan tidak adanya data yang tidak relevan, atau jika bobot data tersebut setara dengan relevansinya terhadap klasifikasi.

Algoritma K-NN adalah salah satu metode yang digunakan untuk analisis klasifikasi, namun beberapa dekade terakhir metode K-NN juga digunakan untuk prediksi. K-NN termasuk kelompok *instance-based learning*. Algoritma ini juga merupakan salah satu teknik *lazy learning*. K-NN dilakukan dengan mencari kelompok k obyek dalam data *training* yang paling dekat (mirip) dengan obyek pada data baru atau data *testing*.

2. METODELOGI PENELITIAN

Agar langkah-langkah yang diambil peneliti dalam penelitian dapat lebih mudah dipahami, berikut ini disertakan penjelasan terperinci mengenai sistematika model metodologi pemecahan masalah.

a. Studi Pustaka

Setelah masalah diperoleh serta telah dianalisa maka akan dilakukan studi kepustakaan yang berguna untuk memperoleh referensi serta literatur dalam menyelesaikan masalah. Studi kepustakaan ini dilakukan dengan cara mencari referensi berupa buku atau jurnal, baik dengan melalui media internet yang berkaitan dengan KNN (K-Nearest Neighbour).

b. Pengumpulan Sample Data /Informasi

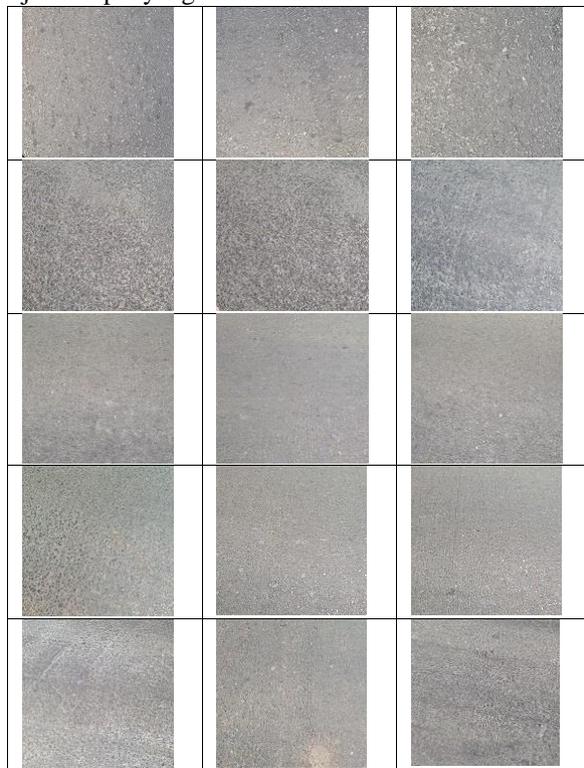
Sampel data pada penelitian ini menggunakan citra yang didapat dari pengambilan gambar menggunakan kamera smartphone dalam bentuk format .jpeg. Subjek dari penelitian ini adalah mengetahui rusak dan tidak rusak jalan raya dengan menggunakan 5 parameter ciri yaitu mean, variance, skewness, kurtosis, dan entropy dengan menggunakan bahasa pemrograman Python. Data dalam penelitian ini menggunakan 30 sampel citra truecolor 24bit dengan format ekstensi *.jpeg.

Berikut adalah sampel foto dari jalan aspal yang diklasifikasi rusak.:



Gambar 1. Sampel jalan aspal rusak.

Sedangkan berikut ini adalah sampel foto dari jalan aspal yang diklasifikasi tidak rusak:



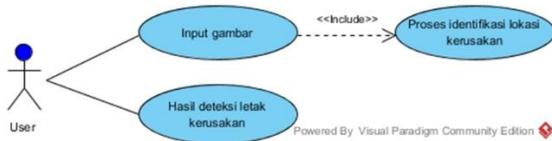
Gambar 2. Sampel jalan aspal bagus.

c. Analisis dan Perancangan Sistem

Analisa dan Perancangan Sistem membahas bagian dalam pembentukan aplikasi seperti metode penggunaan, use case diagram, activity diagram dan penggunaan algoritma dari aplikasi prediksi kerusakan pada gambar jalan aspal.

Secara keseluruhan aplikasi memiliki fitur yaitu: memasukkan gambar (foto) jalan yang akan dideteksi klasifikasinya, melakukan proses identifikasi klasifikasi penentuan apakah gambar jalan tersebut masuk ke klasifikasi jalan aspal rusak atau jalan aspal bagus, dan keluaran simpulan hasil deteksi oleh sistem. Peran pengguna aplikasi ini hanya satu yaitu user aplikasi. Fitur-fitur aplikasi ini bisa dilihat dalam

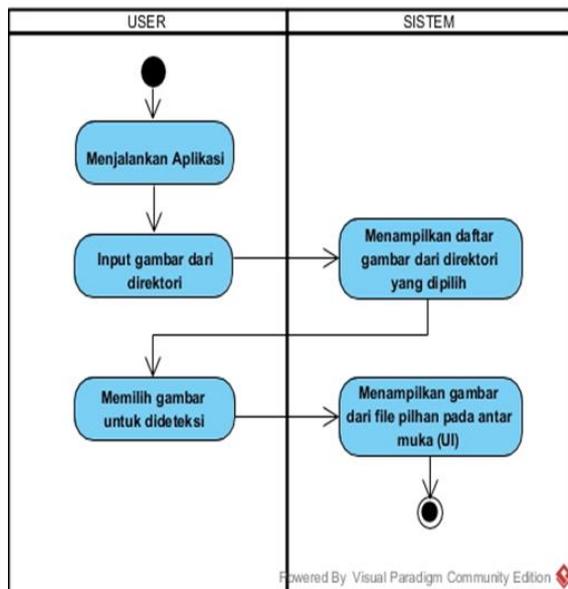
Gambar 3 Use Case Diagram Aplikasi Deteksi Kerusakan Jalan Raya.



Gambar 3. Use Case Diagram Aplikasi Deteksi Kerusakan Jalan Raya.

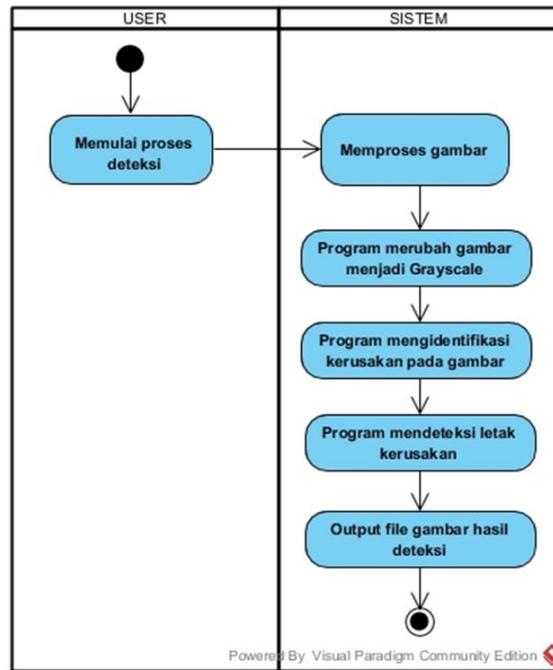
Activity Diagram merupakan diagram yang menggambarkan aktivitas yang terjadi pada sistem. Dari pertama sampai akhir, diagram ini menunjukkan langkah-langkah dalam proses kerja sistem yang kita buat. Berikut dijabarkan activity diagram dari sistem seperti, input gambar, proses pendeteksian objek pada gambar dan output.

Pada Gambar 4 merupakan proses input file gambar ke dalam aplikasi, aplikasi akan menampilkan isi folder dan user akan memilih file gambar yang akan dideteksi ke dalam aplikasi.



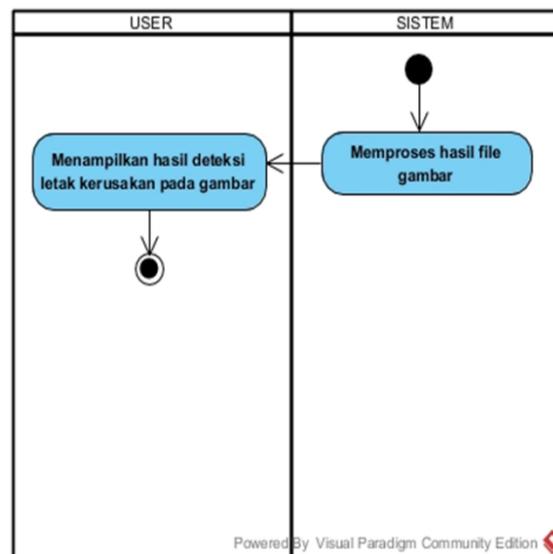
Gambar 4. Activity Diagram Input Gambar.

Pada Gambar 5 proses pendeteksian klasifikasi obyek jalan aspal dilakukan setelah dipilih file gambar dan user memulai proses identifikasi (mendeteksi letak kerusakan). Kemudian sistem akan melakukan proses pada file gambar seperti merubah menjadi keabuan (grayscale) dan mengidentifikasi letak yang rusak pada gambar. Setelah selesai file gambar hasil prediksi akan muncul.



Gambar 5 Activity Diagram proses deteksi file gambar.

Pada Gambar 6 menjelaskan setelah terjadinya proses prediksi dari file gambar, maka hasil deteksi akan ditampilkan dalam bentuk gambar yang pada letak objek rusaknya dikotakan (rectangle).



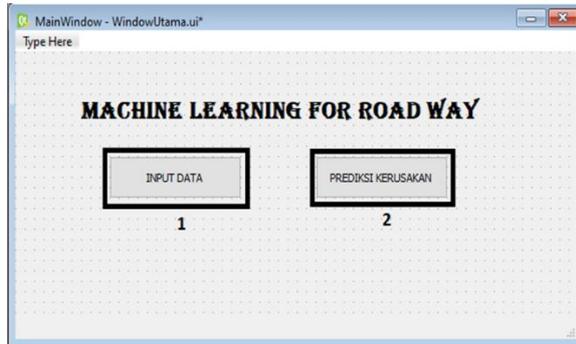
Gabmar 6. Activity Diagram hasil prediksi kerusakan

Pada aplikasi terdapat tampilan utama yang dibuat dengan menggunakan GUI (Graphical User Interface). Berikut penjelasan pada tampilan utama pada aplikasinya.

Tampilan Halaman Utama Home

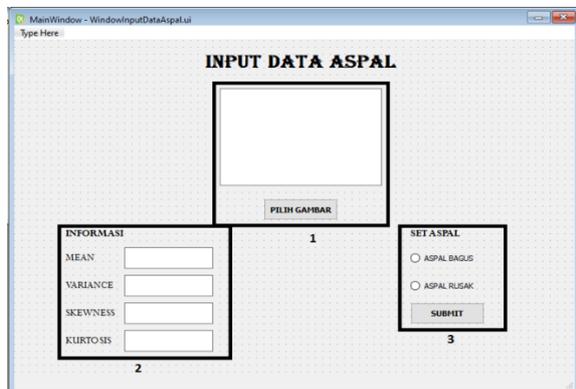
Rancangan tampilan halaman utama di bagi menjadi 2 bagian, yaitu input data sampel dan deteksi kerusakan. Pada Gambar 7 terlihat tampilan antar muka utama atau tampilan awal dari rancangan

aplikasi prediksi kerusakan pada gambar. Pada tampilan ini terdapat beberapa menu properties.



Gambar 7. Tampilan Antar Muka Utama Home.

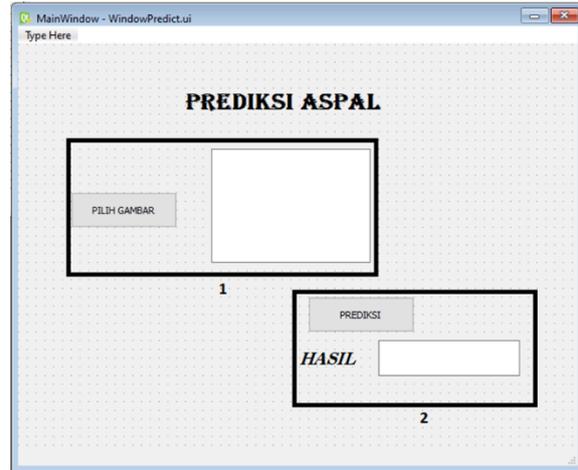
Pada gambar 8 terlihat tampilan input data sampel aplikasi prediksi kerusakan pada gambar. Pada tampilan ini terdapat beberapa menu properties.



Gambar 8. Tampilan Input Sampel Data Jalan Aspal

Pada fitur ini aplikasi akan menerima gambar-gambar sampel jalan aspal sebagai data latih (*data training*) sistem, dimana user menentukan sampel jalan aspal tersebut termasuk jalan rusak atau jalan bagus. Selanjutnya sistem akan menghitung semua parameter histogram gambar dan sistem akan dilatih menggunakan data tersebut.

Berikutnya pada gambar 9 merupakan tampilan fitur sistem yang akan melakukan proses deteksi klasifikasi dari gambar jalan raya yang dimasukkan oleh *user* untuk diuji apakah gambar masukan tersebut termasuk jalan aspal rusak atau jalan aspal bagus.



Gambar 9. Tampilan Prediksi kerusakan jalan.

d. Analisa KNN

Setelah referensi diperoleh maka selanjutnya dilakukan analisa terhadap KNN yaitu mempelajari rumus-rumus dan mulai menghubungkan data yang diperoleh dengan rumus-rumus tersebut sehingga data tersebut menjadi data baru yang akan menghasilkan perhitungan terhadap deteksi kerusakan jalan raya tersebut.

Pada tahap ini algoritma K-NN dalam implementasi pada aplikasi akan menggunakan paket perpustakaan perangkat lunak yang sudah ada.

e. Implementasi (Pelatihan)

Pada tahapan implementasi bertujuan untuk mendapatkan hasil dari analisa tentang deteksi kerusakan jalan raya. Setelah data dianalisa kemudian dilakukan pelatihan untuk mendapatkan hasil yang diinginkan. Pelatihan dilakukan dengan menggunakan dua tahapan. Tahapan-tahapan yang digunakan dalam pelatihan yaitu :

- Pelatihan secara manual

Pelatihan secara manual dilakukan dengan mengolah data yang telah didapatkan dari proses analisa manual. Kemudian data tersebut dimasukan kedalam rumus-rumus perhitungan metode KNN.

- Pelatihan secara software

Data yang telah dianalisa akan dilakukan pelatihan secara software. Pelatihan secara software dilakukan dengan menggunakan software Pycharm dan bahasa pemrograman Python.

f. Pengujian

Pada tahapan pengujian dilakukan untuk mengetahui cara sistem bekerja dalam mengolah data. Proses yang dilakukan yaitu mengolah data. Proses yang dilakukan yaitu mengolah data mentah untuk menjadi data baru dengan menggunakan rumus-rumus KNN dan dilakukan perhitungan secara manual.

3. HASIL DAN PEMBAHASAN

Pembahasan berikut adalah uraian hasil implementasi pada aplikasi deteksi kerusakan pada jalan raya.

a. Implementasi Sistem

Implementasi sistem dilakukan setelah dilakukan analisis dengan mengumpulkan berbagai modul dari aplikasi dan dirangkai ke dalam sistem. Pada tahap ini dilakukan implementasi kepada sistem aplikasi deteksi kerusakan pada gambar.

Implementasi sistem dilakukan untuk membuktikan bahwa aplikasi prediksi kerusakan pada gambar dapat berjalan dengan baik dan berfungsi sesuai dengan yang sudah dirancang pada tahapan sebelumnya. Pada tahap ini pula dilakukan pemeriksaan jika masih ditemukan bug pada aplikasi yang dapat mengganggu jalannya aplikasi. Tampilan utama aplikasi ini adalah seperti terlihat pada gambar 10 tampilan utama aplikasi.



Gambar 10. Tampilan Utama aplikasi.

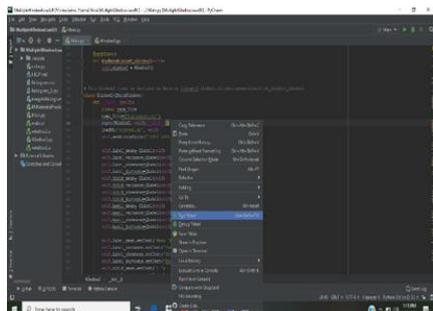
b. Uji Coba Sistem

Uji coba sistem dilakukan dengan melakukan input data dan prediksi kerusakan dengan memilih gambar yang akan diproses.

Uji Coba Input Data

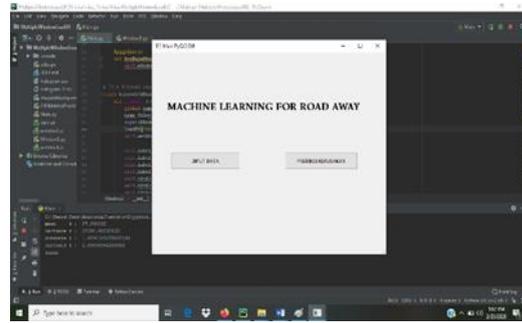
Berikut adalah langkah demi langkah menjalankan aplikasi Input Data hingga berhasil dilakukan.

- Pada halaman source code Main.py klik kanan lalu pilih Run "Main" seperti tampilan gambar 11 di bawah ini.



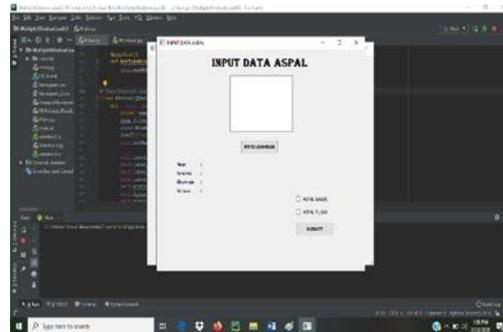
Gambar 11. Tampilan Run Aplikasi

- Setelah klik pilih Run "Main" maka keluar tampilan utama pada aplikasi yang sudah berjalan seperti pada gambar 12, lalu klik Input Data



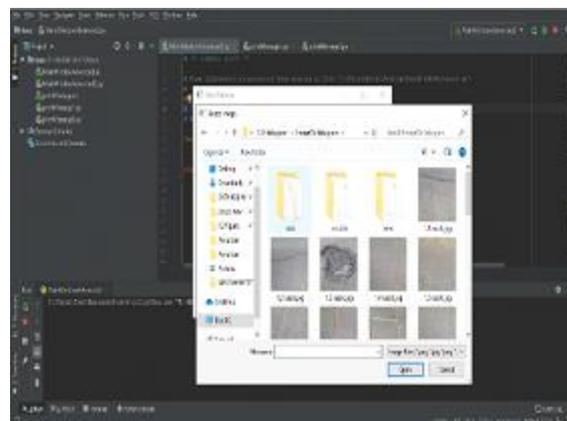
Gambar 12. Tampilan Menu Utama Aplikasi.

- Setelah input Data di klik, maka akan muncul tampilan Input Data Aspal seperti gambar 13 di bawah ini.



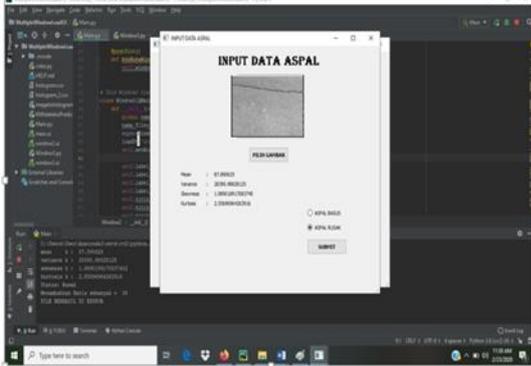
Gabmar 13. Tampilan Input Sampel data aspal.

- Setelah muncul tampilan input data aspal, maka klik pilih gambar untuk memilih gambar mana yang ingin kita input dan ingin mengetahui nilai-nilai parameter dari sebuah gambar tersebut seperti pada gambar 14 di bawah ini. Muncul tampilan open dengan judul Select Image, kemudian cari dan pilih gambar yang diinginkan. Kemudian muncul gambar yang telah dipilih pada tampilan aplikasi Input Data.

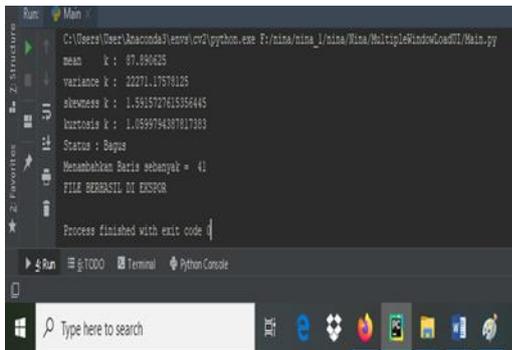


Gambar 14. Pemilihan file dari data gambar.

- Pada gambar 15 muncul tampilan aplikasi dengan gambar yang dipilih dan informasi-informasi untuk nilai parameter mean, variance, skewness dan nilai kurtosis, lalu klik Submit maka data akan tampil di command line seperti gambar 16 dibawah ini.



Gambar 15. Tampilan aplikasi dengan gambar yang dipilih.

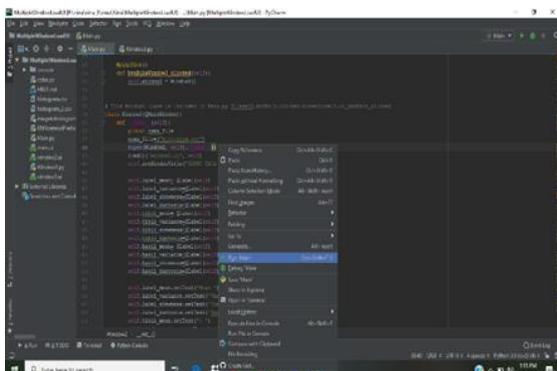


Gambar 16. Tampilan hasil dari Submit input data aspal.

Uji Coba Prediksi Gambar

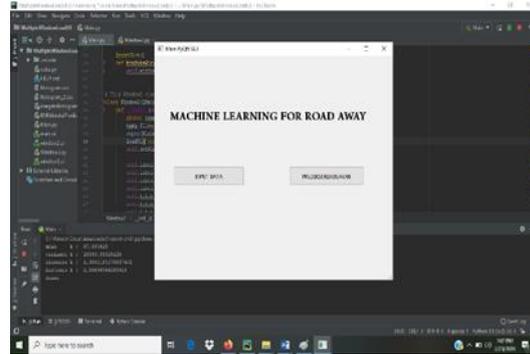
Berikut adalah langkah demi langkah menjalankan aplikasi Prediksi Gambar hingga berhasil dilakukan.

- Pada halaman source code Main.py klik kanan lalu pilih Run “Main” seperti tampilan gambar 17 di bawah ini



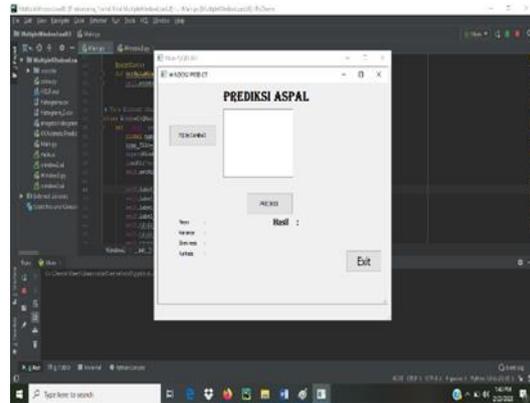
Gambar 17. Tampilan Run aplikasi

- Setelah klik pilih Run “Main” maka keluar tampilan utama pada aplikasi yang sudah berjalan, lalu klik Prediksi gambar 18 dibawah ini.



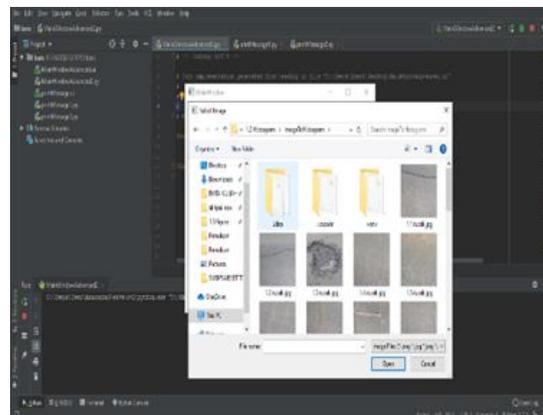
Gambar 18. Tampilan Menu Utama Aplikasi.

- Setelah Prediksi Kerusakan di klik, maka akan muncul tampilan Prediksi Kerusakan seperti gambar 19 di bawah ini.



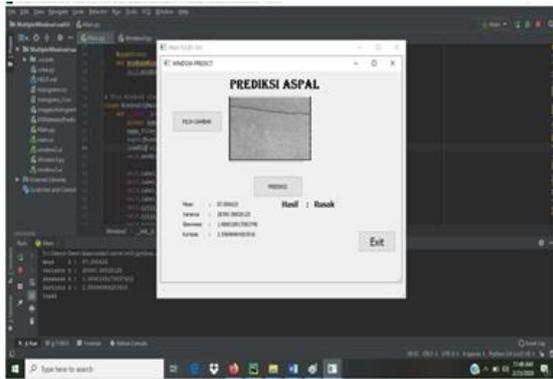
Gambar 19. Tampilan Prediksi Aspal.

- Setelah muncul tampilan prediksi aspal, maka klik pilih gambar untuk memilih gambar mana yang ingin kita input dan ingin mengetahui hasil prediksi serta nilai-nilai parameter dari sebuah gambar tersebut seperti pada gambar 20 di bawah ini.



Gambar 20. Pemilihan file dari daa gambar.

- Muncul tampilan open dengan judul Select Image seperti gambar 20. kemudian cari dan pilih gambar yang diinginkan. Kemudian muncul gambar yang telah dipilih seperti gambar 21 pada tampilan aplikasi Prediksi Data Aspal. Pada gambar 21 merupakan tampilan hasil prediksi gambar rusak dan gambar bagus serta informasi-informasi untuk nilai parameter histogram antara lain *mean, variance, skewness* dan nilai *kurtosis*.



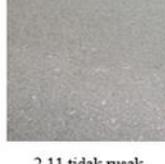
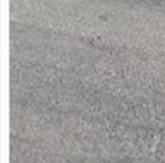
Gambar 21. Tampilan hasil prediksi dari gambar.

Hasil Uji Coba

Hasil uji coba menggunakan 10 sampel citra yang diambil dari hasil foto menggunakan kamera smartphone terhadap objek gambar pada jalan raya, hasil keterangan ada pada tabel sebagai berikut

Tabel 1. Hasil uji coba terhadap sampel citra.

No	Citra Asli	Hasil
1	 1.3 rusak	Terdeteksi dengan tepat
2	 1.5 rusak	Terdeteksi dengan tepat

3	 1.6 rusak	Terdeteksi dengan tepat
4	 1.12 rusak	Terdeteksi dengan tepat
5	 1.14 rusak	Terdeteksi dengan tepat
6	 2.1 tidak rusak	Terdeteksi dengan tepat
7	 2.4 tidak rusak	Terdeteksi dengan tepat
8	 2.7 tidak rusak	Terdeteksi dengan tepat
9	 2.11 tidak rusak	Tidak terdeteksi dengan tepat
10	 2.15 tidak rusak	Terdeteksi dengan tepat

Dimana nilai N adalah jumlah seluruh gambar foto permukaan jalan raya yang diujikan, yaitu terdiri

dari 10 citra dengan label rusak dan tidak rusak, maka dapat dihitung tingkat akurasi sebagai berikut:

Diketahui: N = 10, Hasil yang sesuai = 8, Hasil yang tidak sesuai = 2, maka persentase tingkat akurasi pada aplikasi ini dapat dilihat pada Tabel 2.

Tabel 2. Persentasi Akurasi Sistem.

Data Uji		
Input	Sesuai	Tidak Sesuai
Gambar 1-10	9	1
Akurasi gambar (%)	$= \frac{\text{Jumlah Citra Uji Yang Sesuai}}{N} \times 100\%$ $= \frac{9}{10} \times 100\%$ $= 90\%$	

4. SIMPULAN dan SARAN

Simpulan

Berdasarkan hasil implementasi dan pengujian yang dilakukan terhadap aplikasi maka dapat disimpulkan bahwa :

1. Sistem yang dirancang dengan metode klasifikasi KNN dapat digunakan untuk mendeteksi adanya kerusakan pada permukaan aspal jalan dengan tingkat akurasi 90%.
2. Jumlah data latih sangat mempengaruhi akurasi, jika data latih semakin banyak maka nilai akurasi juga akan semakin naik.
3. Sistem dapat mendeteksi adanya kerusakan pada gambar aspal jalan.
4. Masih terdapat beberapa gambar yang tidak terprediksi sempurna, dikarenakan tercampurnya intensitas cahaya yang tidak seharusnya saat pengambilan gambar.

Saran

Penelitian ini masih dapat dikembangkan, beberapa saran yang dapat dikembangkan untuk penelitian ini sebagai berikut :

1. Membuat jarak dan kondisi pengambilan gambar yang lebih baik dan lebih akurat agar nilai baca dari program dapat lebih baik.
2. Membuat program agar lebih banyak yang menggunakannya, seperti membuat rancang bangun aplikasi berbasis Android.
3. Penelitian selanjutnya dapat dilakukan dengan menggunakan gambar yang diambil dengan kamera thermal.
4. Menyempurnakan tingkat kerusakan dengan banyak jenis kerusakan.

5. DAFTAR PUSTAKA

- [1] Permadi, Yuda. 2015. Aplikasi Pengolahan Citra Untuk Identifikasi Kematangan Mentimun Berdasarkan Tekstur Kulit Buah Menggunakan Metode Ekstrasi Ciri Statistik. Jurnal Informatika: Yogyakarta.
- [2] Saptono, Ristu. 2015. Identifikasi Kualitas Beras dengan Citra Digital. Scientific Journal of Informatics: Solo
- [3] Sutikno. 2017. Classification Of Road Damage From Digital Image Using Backpropagation Neural Network. IJ-AI: Diponegoro.
- [4] Irmawati, Dessy. 2016. Pengolahan Citra Untuk Identifikasi Telur Berdasarkan Ukuran. ELINVO: Yogyakarta.
- [5] Izzah A., Dewi R K., Mutrofin S., 2013. Hybrid Artificial Bee Colony: Penyelesaian Baru Pohon Rentang Berbatas Derajat. (Semnasteknomedia)
- [6] Budiarto, P. Y. (2017). Deteksi Objek Lubang pada Citra Jalan Raya menggunakan Pengolahan Citra Digital, 3(2), 109–118.
- [7] H, A. N., Ichwan, M., & Putra, I. M. S. (2015). Segmentasi Citra Untuk Deteksi Objek Warna Pada Aplikasi Pengambilan Bentuk Citra Rectangle. Jurnal Unpublisher, 1–10.
- [8] Jafar, A., Kadafi, A., & Utamingrum, F. (2018). Deteksi Objek Penghalang Secara Real-Time Berbasis Mobile Bagi Penyandang Tunanetra Menggunakan Analisis Blob, 2(1), 423–432.
- [9] Bay, H., A. Ess, T. Tuytelaars, and L. V. G. (2008). Computer Vision and Image Understanding (CVIU) (p. 1).
- [10] M. Maharlooei, S. Sivarajan, Sreekala G. Bajwa, Jason P. Harmon, J. N. (2017). 75 Detection of soybean aph