ANALISA DATA EKSTRAKSI CIRI CITRA MOMEN HISTOGRAM DAN PERBANDINGAN MODEL ALGORITMA KLASIFIKASI *NAIVE BAYES, NEAREST NEIGHBOR, SUPPORT VECTOR MACHINE,* DAN *DECISION TREE* PADA STUDI KASUS CITRA JALAN ASPAL RUSAK DAN JALAN ASPAL TIDAK RUSAK

EXTRACTION DATA ANALYSIS OF HISTOGRAM MOMENT IMAGE AND COMPARISON OF CLASSIFICATION ALGORITHM MODELS OF NAIVE BAYES, NEAREST NEIGHBOR, SUPPORT VECTOR MACHINE, AND DECISION TREE IN CASE STUDY OF DAMAGED ASPHALT AND UNDAMAGED ASPHALT ROAD IMAGES

Aryo Nur Utomo
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

ABSTRAK
Pertumbuhan teknologi komputer visi membuat banyak pihak terbantu dalam mengotomasi pekerjaan mereka seperti sistem deteksi cerdas pada suatu citra. Deteksi yang dimaksud adalah deteksi kerusakan atau deteksi lubang pada permukaan jalan aspal. Untuk memenuhi hal tersebut, maka dilakukan penelitian dengan mengambil sampel jalan aspal yang diklasifikasi rusak maupun tidak rusak untuk diukur akurasi model algoritma klasifikasi yang terbaik untuk membangun sistem *machine learning* deteksi jalan aspal. Citra sampel foto jalan aspal akan di ekstraksi menjadi momen histogram selanjutnya dataset yang terbentuk akan diterapkan kepada empat algoritma model klasifikasi untuk diukur tingkat akurasi sistemnya. *Tools* yang digunakan dalam penelitian ini adalah *tools library* bahasa pemrograman Pyhton untuk pengolahan citra digital, algoritma klasifikasi, dan menentukan akurasi hasil klasifikasi algoritma tersebut.
Kata Kunci: Klasifikasi, Komputer Visi, Pembelajaran Mesin, Algoritma Klasifikasi, Citra, Python.

ABSTRACT
The growth of computer vision technology has helped many people to automate their work such as an intelligent detection system on an image. Detection is the detection of damage or detection of holes on the surface of the asphalt road. To fulfill this, a research was carried out by taking samples of asphalt roads classified as damaged or not damaged to measure the accuracy of the best classification algorithm model to build a machine learning system for asphalt road detection. The sample image of the asphalt road photo will be extracted into a histogram moment then the formed dataset will be applied to four classification model algorithms to measure the accuracy of the system. The tools used in this study are the Pyhton programming language library tools for digital image processing, classification algorithms, and determining the accuracy of the classification algorithm results
Keywords: Classification, Computer Vision, Machine Learning, Classification Algorithms, Image, Python.

1. PENDAHULUAN

Visi komputer adalah proses bagaimana komputer dapat memperoleh pemahaman tingkat tinggi dari gambar atau video digital. Visi komputer berusaha untuk memahami dan mengotomasikan tugas yang dapat dilakukan oleh sistem visual manusia (wikipedia.org, 2020).

Salah satu pendekatan yang dapat digunakan untuk menganalisis gambar (citra) digital adalah menggunakan ciri statistik orde pertama. Ciri orde pertama didasarkan pada karakteristik histogram citra atau momen histrogram. Ciri orde pertama (momen histogram) digunakan untuk membedakan tekstur makrostruktur (perulangan pola lokal secara periodik). Ekstraksi ciri citra orde pertama antara lain *mean, variance, skewness, kurtosis*.

Studi kasus pada penelitian ini adalah sampel citra (foto) digital kondisi jalan aspal yang diambil menggunakan kamera *smartphone* dan diklasifikasi secara visual oleh peneliti menjadi dua kelas klasifikasi yaitu jalan aspal rusak dan jalan aspal tidak rusak.

Klasifikasi termasuk kedalam *machine learning supervised learning* karena menggunakan sekumpulan data untuk dianalisis

terlebih dahulu, kemudian pola dari hasil analisis tersebut digunakan untuk pengklasifikasian data uji atau untuk memprediksi data baru (*unseen*).

Teknik klasifikasi dibagi menjadi lima kategori berdasarkan perbedaan konsep matematika, yaitu berbasis statistik, berbasis jarak, berbasis pohon keputusan, berbasis jaringan syaraf, dan berbasis *rule* (Karim.M, 2013). Ada banyak algoritma dari masing-masing kategori tersebut, namun yang populer dan sering digunakan diantaranya yaitu *naive bayes*, *nearest neighbour*, *support vector machine* dan *decision tree*.

Pada penelitian ini peneliti akan menganalisis dan membandingkan algoritma klasifikasi tersebut pada data ciri citra orde pertama terhadap sampel citra (foto) digital aspal jalan yang akan dibandingkan berdasarkan tingkat akurasinya. Sehingga informasi hasil penelitian ini dapat digunakan untuk membangun model *machine learning* yang terbaik untuk menentukan klasifikasi data baru citra aspal.

Peneliti menggunakan *tools library* bahasa pemrograman Pyhton untuk pengolahan citra digital, algoritma klasifikasi, dan menentukan akurasi hasil klasifikasi algoritma tersebut.

2. METODE PENELITIAN

Dalam penelitian ini dilakukan langkah-langkah dimulai dari studi literatur, pengambilan sampel data foto, kemudian melakukan *preprocessing* sampel data, selanjutnya memproses sampel-sampel data.

Studi Literatur

Dalam bagian ini dipaparkan berbagai informasi yang berhubungan dengan penelitian dan perancangan program.

- **Visi komputer (*Computer Vision*)**

  Visi komputer merupakan kumpulan dari metode-metode untuk mendapatkan, memproses, menganalisis suatu gambar atau dalam arti lain visi komputer, merupakan kumpulan metode-metode yang digunakan untuk menghasilkan angka-angka atau simbol-simbol yang didapat dari gambar yang diambil dari dunia nyata agar komputer dapat mengerti dan mengolah apa makna dari gambar tersebut.

- **Pengolahan Citra**

  Bertujuan memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau mesin(dalam hal ini komputer). Teknik-teknik pengolahan citra mentransformasikan citra menjadi citra lain. Jadi, masukannya adalah citra dan keluarannya juga citra, namun citra keluaran mempunyai kualitas lebih baik daripada citra masukan. Termasuk ke dalam bidang ini juga adalah pemampatan citra (image compression).

- **Pengenalan Pola**

  Pengenalan Pola (*Pattern Recognition*), bidang ini berhungan dengan proses indenfikasi obyek pada citra atau interpretasi citra. Proses ini bertujuan untuk mengekstrak informasi/pesan yang disampain oleh gambar citra.

- **Histogram Citra**

  Histogram citra adalah grafik yang menggambarkan penyebaran nilai-nilai intensitas pixel dari suatu citra atau bagian tertentu di dalam citra. Histogram citra (image histogram) merupakan informasi yang penting mengenai isi citra digital. Dari sebuah histogram dapat diketahui frekuensi kemunculan nisbi (relative) dari intensitas pada citra tersebut. Histogram juga dapat menunjukkan banyak hal tentang kecerahan (brightness) dan kontas (contrast) dari sebuah gambar. Karena itu, histogram adalah alat bantu yang berharga dalam pekerjaan pengolahan citra baik secara kualitatif maupun kuantitatif. Histogram citra merupakan diagram yang menggambarkan distribusi frekuensi nilai intensitas piksel dalam suatu citra. Sumbu horizontal merupakan nilai intensitas piksel sedangkan sumbu vertikal merupakan frekuensi/jumlah piksel.

- **Klasifikasi**

  Klasifikasi adalah salah satu pembelajaran yang paling umum di *machine learning*. Klasifikasi didefinisikan sebagai bentuk analisis data untuk mengekstrak model yang akan digunakan untuk memprediksi label kelas (J. Iawe. Han., etal, 2012). Kelas dalam klasifikasi merupakan atribut dalam satu set data yang paling unik yang merupakan variabel bebas dalam statistik (P. Yoo, etal, 2012). Klasifikasi data terdiri dari dua proses yaitu tahap pembelajaran dan tahap pengklasifikasian. Tahap pembelajaran merupakan tahapan dalam pembentukan model klasifikasi, sedangkan tahap pengklasifikasian merupakan tahapan penggunaan model klasifikasi untuk memprediksi label kelas dari suatu data. Contoh sederhana dari teknik *machine learning* klasifikasi adalah pengklasifikasian hewan berdasarkan atribut jumlah kaki, habitat dan organ pernafasannya akan diklasifikasikan ke

dalam dua label kelas yaitu unggas dan ikan. Label kelas unggas adalah data yang memiliki jumlah kaki dua, habitatnya di darat, dan organ pernafasannya menggunakan paru-paru, sedangkan label kelas ikan adalah data yang memiliki jumlah kaki nol (tidak memiliki kaki), habitat di air, dan organ pernafasannya menggunakan insang. Banyak algoritma yang dapat digunakan dalam pengklasifikasian data, namun dalam penelitian ini hanya akan membandingkan tiga algoritma saja, yakni naive bayes, nearest neighbour, dan decision tree.

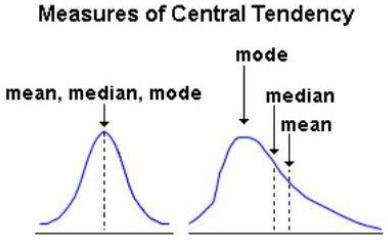
- **Ekstraksi Ciri Statistik pada Citra**

  Suatu proses klasifikasi citra berbasis analisis tekstur pada umumnya membutuhkan tahapan ekstraksi ciri, yang terdiri dari tiga macam metode salah satunya metode statistik. Metode statistik menggunakan perhitungan statistik distribusi derajat keabuan (histogram) dengan mengukur tingkat kekontrasan, granularitas, dan kekasaran suatu daerah dari hubungan ketetanggaan antar piksel di dalam citra. Paradigma statistik ini penggunaannya tidak terbatas, sehingga sesuai untuk tekstur-tekstur alami yang tidak terstruktur dari sub pola dan himpunan aturan (mikrostruktur). Dari nilai-nilai pada histogram yang dihasilkan dapat dihitung beberapa parameter ciri, antara lain adalah *mean, variance, skewness,* dan *kurtosis.*

  - **Mean (μ)**

    Salah satu ukuran statistik yang menunjukkan ukuran tendensi sentral atau nilai pusat data pengamatan.

$$\mu = \sum_{n=0}^{N} fn \; P(fn) \qquad (1)$$

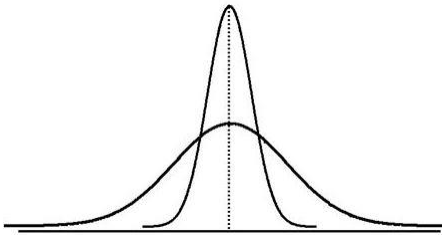


Gambar 1. Grafik tendensi sentral (gurubelajarku.com, 2020).

  - **Variance (σ2)**

    Ukuran dispersi (variasi) dalam statistik yang mengukur sejauh mana suatu distribusi ditarik atau dipencar.

$$\sigma^2 = \sum_{n=0}^{N} (fn - \mu)^2 \; P(fn) \qquad (2)$$

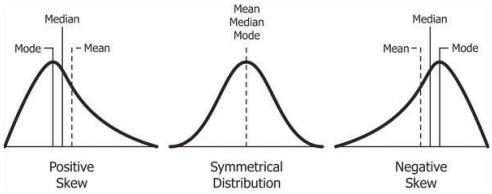


Gambar 2. Grafik dispersi (gurubelajarku.com, 2020).

  - **Skewness (α3)**

    Skewness adalah derajat ketidaksimetrisan suatu distribusi. Skewness (ukuran kemiringan) merupakan suatu bilangan yang dapat menunjukan miring atau tidaknya bentuk kurva suatu distribusi frekuensi. Jika kurva frekuensi suatu distribusi memiliki ekor yang lebih memanjang ke kanan (dilihat dari *mean*nya) maka dikatakan menceng kanan (positif) dan jika sebaliknya maka menceng kiri (negatif).

$$\alpha_3 = \frac{1}{\sigma^3} \sum_{n=0}^{N} (fn - \mu)^3 \; P(fn) \qquad (3)$$

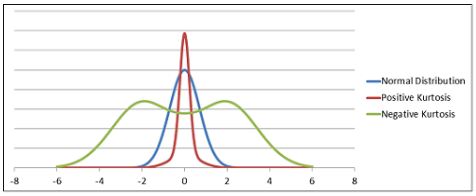


Gambar 3. Grafik skewness (gurubelajarku.com, 2020).

  - **Kurtosis (α4)**

    Keruncingan atau kurtosis adalah tingkat ketinggian puncak atau keruncingan dari sebuah distribusi yang biasanya diambil secara relatif terhadap suatu distribusi normal. Berdasarkan keruncingannya, kurva distribusi dapat dibedakan atas tiga macam, yaitu (1) Leptokurtik merupakan distribusi yang memiliki puncak relatif tinggi; (2) Platikurtik merupakan distribusi yang memiliki puncak hampir mendatar; (3) Mesokurtik merupakan distribusi yang memiliki puncak tidak tinggi dan tidak mendatar. Bila distribusi merupakan distribusi simetris, maka distribusi mesokurtik dianggap sebagai distribusi normal.

$$\alpha_4 = \frac{1}{\sigma^4} \sum_{n=0}^{N} (fn - \mu)^4 \; P(fn) - 3 \qquad (4)$$



Gambar 4. Grafik kurtosis (gurubelajarku.com, 2020).

- **Algoritma Naive Bayes**

  Teorema bayes adalah perhitungan statistik dengan menghitung probabilitas kemiripan kasus lama yang ada dibasis kasus dengan kasus baru. Naive bayes termasuk ke dalam pembelajaran *supervised*, sehingga pada tahapan pembelajaran dibutuhkan data awal berupa data pelatihan untuk dapat mengambil keputusan. Pada tahapan pengklasifikasian akan dihitung nilai probabilitas dari masing-masing label kelas yang ada terhadap masukan yang diberikan. Label kelas yang memiliki nilai probabilitas paling besar yang akan dijadikan label kelas data masukan tersebut. Naive bayes merupakan perhitungan teorema bayes yang paling sederhana, karena mampu mengurangi kompleksitas komputasi menjadi multiplikasi sederhana dari probabilitas. Selain itu, algoritma naive bayes juga mampu menangani set data yang memiliki banyak atribut (P. Yoo, etal, 2012). Persamaan dari naive bayes sebagai berikut:

$$P(C_i \mid X) = \frac{P(X \mid C_i)P(C_i)}{P(X)} \quad (5)$$

  Keterangan :
  X : Kriteria suatu kasus berdasarkan masukan
  C_i : Kelas solusi pola ke-i, dimana i adalah jumlah label kelas
  $P(C_i|X)$: Probabilitas kemunculan label kelas C_i dengan kriteria masukan X
  $P(X|C_i)$: Probabilitas kriteria masukan X dengan label kelas C_i
  $P(C_i)$: Probabilitas label kelas C_i

- **Algoritma Nearest Neighbour**

  Nearest Neighbour adalah algoritma pengklasifikasian yang didasarkan pada analogi,
  yaitu membandingkan data uji dengan data pelatihan yang berada dekat dengan dan memiliki
  kemiripan dengan data uji tersebut (R. Entezari-Maleki). Kemiripan data uji dengan data pelatihan didasarkan pada jaraknya. Banyak persamaan yang dapat digunakan untuk menghitung jarak antara data uji dan data pelatihan.

  Pendekatan perhitungan jarak/kemiripan yang umum digunakan untuk atribut yang bertipe numerik, yaitu *euclidean distance* (Entezari-Maleki. R) dengan persamaan berikut:

$$Dist(x1, x2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})} \quad (6)$$

  Keterangan:
  n : jumlah data
  x1 : data uji
  x2 : data pembelajaran
  Persamaan yang kedua yaitu *Manhattan distance* [11] sebagai berikut:

$$Dist(p_i(an), p_i(nc)) = \frac{p_i(an) - p_i(nc)}{\max\_dist\_i} \quad (7)$$

  Keterangan:
  pi : atribut ke-i
  an : data pembelajaran
  nc : data uji

- **Algoritma Decision Tree**

  Algotirma decision tree merupakan algoritma yang umum digunakan untuk pengambilan keputusan. Decision tree akan mencari solusi permasalahan dengan menjadikan kriteria sebagai node yang saling berhubungan membentuk seperti struktur pohon (S. H. Babic, etal, 2000). Decision tree adalah model prediksi terhadap suatu keputusan menggunakan struktur hirarki atau pohon (N. Jayanti, etal, 2008). Setiap pohon memiliki cabang, cabang mewakili suatu atribut yang harus dipenuhi untuk menuju cabang selanjutnya hingga berakhir di daun (tidak ada cabang lagi). Konsep data dalam decision tree adalah data dinyatakan dalam bentuk tabel yang terdiri dari atribut dan record. Atribut digunakan sebagai parameter yang dibuat sebagai kriteria dalam pembuatan pohon.

  Proses dalam decision tree adalah sebagai berikut (V. Mandasari and B. A. Tama):

  1. Mengubah bentuk data (tabel) menjadi model pohon :
     Hal yang dilakukan pada tahapan ini adalah menentukan atribut yang terpilih mulai dari akar, cabang hingga menuju keputusan. Banyak pendekatan yang dapat digunakan untuk menentukan atribut terpilih, pada penelitian ini akan menggunakan perhitungan gainratio dari setiap kriteria dengan data sampel. Untuk menghitung nilai gainratio dapat dilakukan dengan persamaan sebagai berikut:

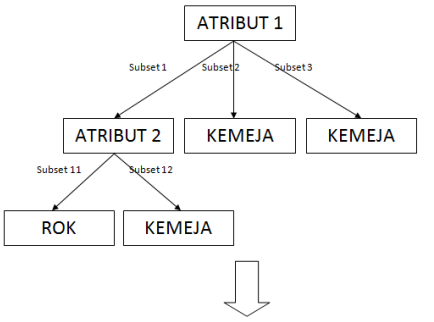
$$Gainratio(S,A) = \frac{Gain(S,A)}{SplitInformation(S,A)} \quad (8)$$

Dimana nilai information gain bermakna seberapa banyak informasi yang diperoleh dengan mengetahui nilai suatu atribut, sedangkan nilai split information digunakan untuk suatu atribut yang memiliki banyak instance (lebih dari dua dan beragam).

2. Mengubah model pohon menjadi *rule*: Formula untuk membangkitkan *rule* didefinisikan sebagai berikut:

IF premis THEN konklusi (9)

Simpul akar dan cabang akan menjadi premis dari aturan, sedangkan simpul daun akan menjadi bagian dari konklusinya (solusi). Tiap premis yang terdapat dalam satu atribut akan dihubungkan dengan hubungan disjungsi, sedangkan premis yang memiliki lanjutan premis pada cabang selanjutnya akan dihubungkan dengan konjungsi.



Gambar 4. Proses Model Pohon Menjadi *Rule* (.....)

3. Menyederhanakan *rule* (*Pruning*)
Pada proses penyederhanaan *rule*, tahapan tahapan dilakukan sebagai berikut:
   1. Membuat tabel distibusi terpadu dengan menyatakan semua nilai kejadian pada setiap *rule*.
   2. Menghitung tingkat independensi antara kriteria pada suatu *rule*, yaitu antara atribut dengan target atribut (perhitungan tingkat independensi menggunakan *test of independency Chi-Square*).
   3. Mengeliminasi kriteria yang dianggap tidak perlu, yaitu yang memiliki tingkat independensi tinggi.

   Misalkan yang ingin dilihat adalah pengaruh jenis pakaian terhadap penentuan solusi pola pakaian yang dapat dibuat, tentukan terlebih dahulu tingkat signifikansinya (λ = 0.05), sehingga dapat dihitung *degree of freedom* dengan persamaan berikut:

   {0.05;(r-1)*(c-1)} (10)

   Keterangan:
   r : jumlah baris
   c : jumlah kolom
   Setelah diperoleh nilainya maka dapat dilihat pada tabel untuk memperoleh nilai X^2_{tabel} untuk dibandingkan dengan X^2_{hitung}. X^2_{hitung} diperoleh melalui persamaan berikut:

$$X^2_{hitung} = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{(n_{ij} - e_{ij})^2}{e_{ij}} \quad (11)$$

   Keterangan:
   n_{ij} : nilai *record* baris ke i kolom ke j dari tabel distribusi terpadu.
   Sedangkan nilai e_{ij} diperoleh melalui persamaan berikut:

$$e_{ij} = \frac{n_i \bullet n_j}{n} \quad (12)$$

   Keterangan:
   ni : marjinal dari baris ke i
   nj : marjinal dari kolom ke j
   n : jumlah record data
   Jika nilai $X^2_{hitung} \leq X^2_{tabel}$ artinya atribut tersebut tidak mempengaruhi atribut target, sehingga *rule* dari atribut tersebut dapat dihilangkan. Namun sebaliknya jika nilai $X^2_{hitung} > X^2_{tabel}$ berarti atribut tersebut mempengaruhi atribut target, sehingga rule dari atribut tersebut tidak dapat dihilangkan.

- **Algoritma Support Vector Machine**

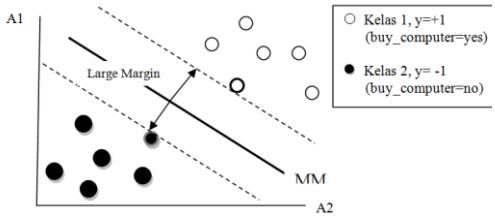
Algoritma Support Vector Machine (SVM) sebagai metode baru yang menjanjikan untuk klasifikasi data, baik linier maupun nonlinier. Langkah awal suatu algoritma SVM adalah pendefinisian persamaan suatu *hyperplane* pemisah yang dituliskan dengan :

$$W.X + b = 0 \quad (13)$$

W adalah suatu bobot vektor, yaitu W = {W_1, W_2,…,W_n}; n adalah jumlah atribut dan b merupakan suatu skalar yang disebut dengan bias. Jika berdasarkan pada atribut A1, A2 dengan permisalan tupel pelatihan X = (x1, x2), x1 dan x2 merupakan nilai dari atribut A1 dan A2, dan jika b dianggap sebagai suatu bobot tambahan w_0, maka persamaan suatu *hyperplane* pemisah dapat ditulis ulang sebagai berikut:

$$w_0 + w_1 x_1 + w_2 x_2 = 0 \quad (14)$$

Setelah persamaan dapat didefinisikan, nilai x_1 dan x_2 dapat dimasukkan ke dalam persamaan untuk mencari bobot w_1, w_2, dan w_0 atau b.



Gambar 5. Pemisahan dua kelas data dengan margin maksimum

Pada Gambar 5, SVM menemukan *hyperplane* pemisah maksimum, yaitu *hyperplane* yang mempunyai jarak maksimum antara tupel pelatihan terdekat. *Support vector* ditunjukkan dengan batasan tebal pada titik tupel. Dengan demikian, setiap titik yang terletak di atas *hyperplane* pemisah memenuhi rumus:

$$w_0 + w_1 x_1 + w_2 x_2 > 0 \quad (15)$$

Sedangkan, titik yang terletak di bawah *hyperplane* pemisah memenuhi rumus:

$$w_0 + w_1 x_1 + w_2 x_2 < 0 \quad (16)$$

Melihat dua kondisi di atas, maka didapatkan dua persamaan hyperplane yaitu:

H_1: $w_0 + w_1x_1 + w_2x_2 \geq 1$ untuk $y_i = +1$ (17)

H_2: $w_0 + w_1x_1 + w_2x_2 \leq -1$ untuk $y_i = -1$ (18)

Perumusan model SVM menggunakan trik matematika yaitu formula *Lagrangian*. Berdasarkan *Lagrangian formulation*, Maksimul Margin *Hyperplane* (MMH) dapat ditulis ulang sebagai suatu batas keputusan (*decision boundary*) yaitu:

$$d(X^T) = \sum_{i=1}^{l} y_i \alpha_i X_i X^T + b_0 \quad (19)$$

yi adalah label kelas dari support vector X_i. X^T merupakan suatu tupel test. α_i dan b_0 adalah parameter numerik yang ditentukan secara otomatis oleh optimalisasi algoritma SVM dan l adalah jumlah *vector support*.

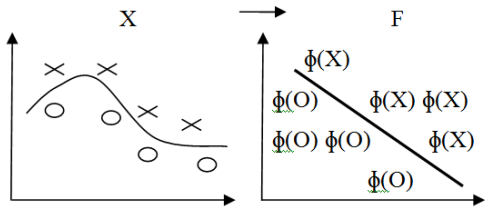
Adanya *hyperplane* yang maksimum mampu memberikan akurasi yang lebih baik pada data yang dapat dipisahkan secara *linier*, namun hal tersebut tidak berlaku bagi data yang tidak dapat dipisahkan secara *linier* [25]. Model pembelajaran SVM memperkenalkan istilah penalti untuk klasifikasi kesalahan dalam fungsi objektif dengan menggunakan parameter biaya [14]. Dengan adanya parameter biaya terhadap kesalahan, maka fungsi optimasi SVM menjadi:

$$min \frac{1}{2}|W|^2 + C\sum_{i=1}^{m} \xi_i \quad (20)$$

$\xi i \geq 0, 1 \leq m \leq l$ merupakan variabel *slack* untuk memungkinkan kesalahan beberapa klasifikasi dan C yang disebut sebagai parameter biaya untuk mengontrol keseimbangan antara margin dan kesalahan klasifikasi [13]. Dengan demikian pembatas pada dua kelas diberi suatu tambahan berupa variable slack ξi sehingga margin pembatas menjadi:

$$x_i w + b \geq +1 - \xi_i \text{ untuk } y_i = +1 \quad (21)$$

$$x_i w + b \leq -1 + \xi_i \text{ untuk } y_i = -1 \quad (22)$$



Gambar 6. Suatu kernel mengubah *problem* yang tidak linear menjadi linear dalam ruang baru.

Pada Gambar 6 memperlihatkan adanya permasalahan klasifikasi tidak dapat diselesaikan secara *linear* pada sampel data X. Pengubahan dari *problem* data non *linear* ke *linear* membutuhkan hitungan yang komplek. Maka diperlukan trik matematika lain yang dapat mempermudah perhitungan, dalam hal ini suatu penggunaan kernel mulai diterapkan. Terdapat 3 persamaan pada kernel SVM yang dapat digunakan yaitu:

a. Polinomial Kernel
b. Kernel Berbasis Gaussian Radial (RBF)
c. Sigmoid Kernel

Salah satu kernel yang populer digunakan di SVM adalah kernel RBF, yang memiliki parameter yang dikenal sebagai *Gaussian width*, σ [11]. Sangat berbeda dengan RBF *Network*, SVM dengan kernel RBF atau biasa disingkat *RBFSVM* dapat secara automatis menentukan jumlah dan lokasi dari pusat dan nilainilai bobot.

- **Scikit-learn *Library***

Scikit-learn atau Sklearn adalah paket *library* bahasa python yang dibangun diatas

paket *library* NumPy, SciPy, dan Matplotlib. Sklearn mampu melakukan pemrosesan data ataupun melakukan *training* data untuk kebutuhan *machine learning*.

Ada banyak fitur yang dapat digunakan dengan sklearn ini, seperti *Classification, Regression, Clustering, Dimensionality reduction, Model selection, dan Preprocessing data*, dengan didalamnya telah terdapat algoritma-algoritma untuk klasifikasi maupun klusterisasi data. Sklearn digunakan untuk membangun model *machine learning* baik *Supervised learning* maupun *Unsupervised learning*.

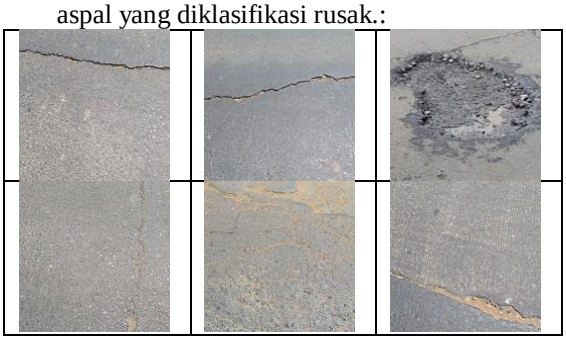
- **OpenCV *Library***

  OpenCV adalah suatu library gratis yang dikembangkan oleh developer-developer Intel Corporation. Library ini terdiri dari fungsi-fungsi computer vision dan API (Aplication Programming Interface) untuk image processing high level maupun low level dan sebagai optimasi aplikasi realtime. OpenCV sangat disarankan untuk programmer yang akan berkutat pada bidang computer vision, karena library ini mampu menciptakan aplikasi yang handal, kuat dibidang digital vision, dan mempunyai kemampuan yang mirip dengan cara pengolahan visual pada manusia, Karena library ini bersifat cuma-cuma dan sifatnya yang open source, maka dari itu OpenCV tidak dipesan khusus untuk pengguna arsitektur Intel, tetapi dapat dibangun pada hamper semua arsitektur.

Pengambilan Sampel Citra Jalan Aspal

Sampel foto (citra) jalan aspal diambil diambil menggunakan kamera *smartphone* dengan resolusi 8 mega pixel. Dan disimpan dalam format digital file jpg. Foto jalan aspal diambil sejumlah 30 sampel foto yang dibagi menjadi 15 foto diklasifikasi termasuk jalan aspal rusak dan 15 foto diklasifikasi termasuk jalan aspal tidak rusak.

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



Gambar 7. Sampel jalan aspal rusak.

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

Gambar 8. Sampel jalan aspal tidak rusak.

Histogram Sampel Foto

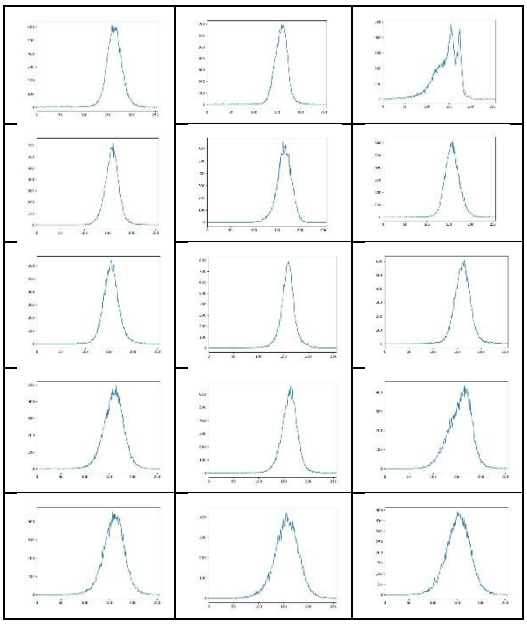
Sampel foto awal hasil pengambilan kamera memiliki *channel color* (RGB) atau 3 *channel*. Berdasarkan pertimbangan bahwa sampel foto aspal memiliki lebih banyak piksel hitam dan putih maka dari sampel foto yang ada masing-masing akan di konversi menjadi hanya memiliki *channel grayscale* yang memiliki 1 *channel*. Keuntungan lain jika foto dalam bentuk *grayscale* adalah proses komputasi komputer yang jauh lebih ringan.

Selanjutnya dari hasil konversi foto dari RGB ke *grayscale* masing-masing akan

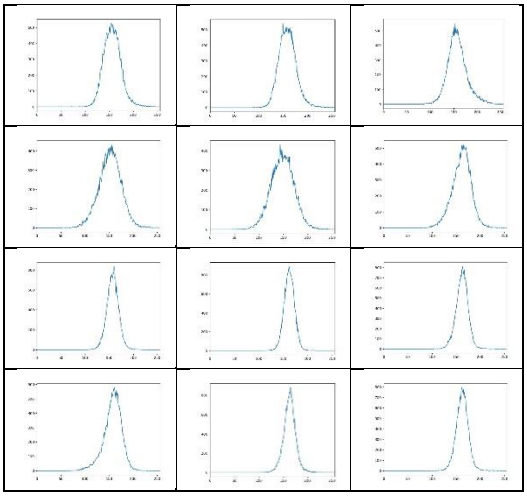
dibuatkan grafik histogram citra, dengan kode program berikut:

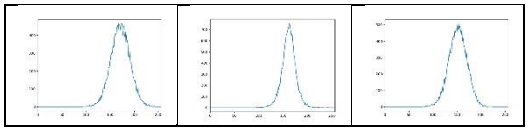
```
# membaca input dari gambar
filepath = "foto/"
filename = "tidakrusak15.jpg"
gambar = cv2.imread(filepath+filename, 0)

# dapatkan frekuensi dari piksel dalam range 0-255
menggunakan opencv
histogram = cv2.calcHist([gambar], [0], None, [256], [0,
256])

# menampilkan grafik plotting dari gambar
plt.plot(histogram)
plt.xlim([0, 256])
plt.show()
plt.savefig(filepath+"hist"+filename)
exit(0)
```

Berikut ini adalah grafik histogram grayscale dari masing-masing citra. Baik untuk foto jalan aspal rusak maupun untuk foto jalan aspal tidak rusak.



Gambar 9. Histogram jalan aspal rusak.





Gambar 10. Histogram jalan aspal tidak rusak.

Tabel Momen Histogram

Selanjutnya dari hasil histogram setiap gambar akan dihitung momen histogram yang merupakan ekstraksi ciri citra dengan 1 *channel*. Momen histogram citra yang diukur adalah *mean*, *variance*, *skewness*, dan *kurtosis*.

Kode program untuk menghitung momen histogram citra adalah seperti berikut:

```
# membaca input dari gambar
filepath = "foto/"
filename = "tidakrusak15.jpg"
gambar = cv2.imread(filepath + filename, 0)

# dapatkan frekuensi dari piksel dalam range 0-255
menggunakan opencv
histogram = cv2.calcHist([gambar], [0], None, [256], [0,
256])

# mengekstraksi ciri histogram image untuk
mendapatkan mean, varian, skewness, dan kurtosis
nilai_mean = np.mean(histogram)
nilai_variance = np.var(histogram)
nilai_skewness = skew(histogram)
nilai_kurtosis = kurtosis(histogram)

try:
   filecsv = open("tabelaspal.csv", 'a')
   # membuat tabel perhitungan moment histogram citra
   # memiliki kolom mean, variance, skewness, kurtosis
   filecsv.write(
      str(nilai_mean) + "," + str(nilai_variance) + "," +
      str(nilai_skewness[0]) + "," + str(nilai_kurtosis[0])
+ "," + "1" +
      "\n")
   filecsv.close()
except IOError as error:
   print("Proses tulis file gagal karena: ", error)

exit(0)
```

Hasil perhitungan momen histogram foto aspal untuk keseluruhan sampel data dengan klasifikasi rusak dan klasifikasi tidak rusak adalah seperti pada tabel 1 berikut.

Tabel 1. Hasil momen histogram citra

No	Mean	Variance	Skewness	Kurtosis	Label
1	87.890.625	28.390.941	19.890.262	25.569.854	0
2	87.890.625	33.910.215	21.813.004	33.737.946	0
3	87.890.625	15.383.617	14.245.489	0.9126375	0
4	87.890.625	32020.35	2.169.597	3.439.227	0
5	87.890.625	28.933.395	19.896.824	26.201.334	0
6	87.890.625	26.945.678	1.910.658	23.258.834	0
7	87.890.625	28.328.152	19.995.879	26.795.568	0
8	87.890.625	35.967.285	2.494.721	51.237.307	0
9	87.890.625	27.133.764	19.225.085	23.078.408	0
10	87.890.625	20010.47	15.426.179	0.94011736	0
11	87.890.625	28.915.637	20.364.401	2.842.358	0
12	87.890.625	16.468.367	13.466.499	0.39654398	0
13	87.890.625	18.502.547	14.549.702	0.6299615	0
14	87.890.625	15.955.501	1.303.954	0.21832776	0
15	87.890.625	14.970.258	12.265.476	0.01307106	0
16	87.890.625	24.501.598	16.910.743	1.316.504	1
17	87.890.625	24.797.785	17.310.286	14.894.028	1
18	87.890.625	21546.13	17.356.443	16.975.527	1
19	87.890.625	17165.0	13.596.833	0.38620543	1
20	87.890.625	16.169.962	12.569.983	0.05737543	1
21	87.890.625	22.117.543	1.674.978	14.450.965	1
22	87.890.625	39.014.082	2.385.064	44.702.954	1
23	87.890.625	45.224.047	2.543.929	52.011.375	1
24	87.890.625	38797.72	2.393.977	4.504.081	1
25	87.890.625	24.856.865	18.656.343	21.544.833	1
26	87.890.625	40.561.332	25.245.905	52.686.033	1
27	87.890.625	38756.59	23.687.177	4.355.522	1
28	87.890.625	20.697.885	15.270.827	0.85372806	1
29	87.890.625	35.107.816	23.144.014	4.151.848	1
30	87.890.625	22.271.176	15.915.728	10.599.794	1

Pada tabel 1 diatas untuk kolom Label maka yang bernilai 0 adalah jalan aspal berklasifikasi rusak, sedangkan yang bernilai 1 adalah jalan aspal berklasifikasi tidak rusak. Tabel momen histogram tersebut menjadi dataset yang akan dipakai untuk membuat model *supervised learning* masing-masing menggunakan algoritma klasifikasi *Naive Bayes*, *Nearest Neighbor*, *SVM*, dan *Decision Tree*.

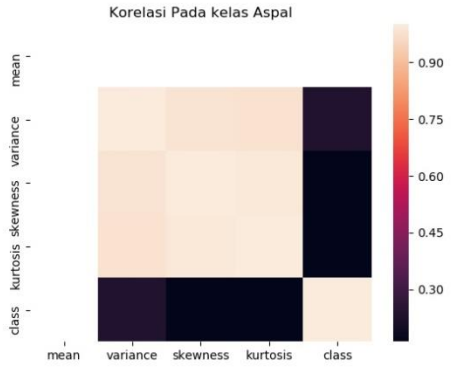
3. HASIL DAN PEMBAHASAN

Pembahasan berikut adalah uraian hasil pemrosesan sampel data foto aspal dan pembahasan dataset yang terbentuk serta penerapan algoritma klasifikasi pada dataset tersebut.

Analisis Data Momen Histogram

Unutk menganalisis fitur dataset hasil ekstraksi ciri citra berupa momen histogram yaitu *mean*, *variance*, *skewness*, dan *kurtosis* maka digunakan *tools heatmap*. *Heatmap* menggambarkan tingkat korelasi dari fitur-fitur dataset yang memberikan seberapa kuat suatu fitur akan membedakan pada label datasetnya. Berikut kode dan *heatmap* dataset citra jalan aspal.

```
# menentukan nama-nama kolom
colnames = ["mean", "variance", "skewness", "kurtosis",
"class"]

# membaca dataset menjadi dataframe dgn pandas
dataset = pd.read_csv("tabelaspal.csv", sep=","
,header=None, names=colnames)

# cek dgn print dataframe
#print(dataset.head())

# menganalisis dataset. Melihat korelasi data dengan
heatmap
plt.figure(1)
sns.heatmap(dataset.corr())
plt.title("Korelasi Pada kelas Aspal")
plt.show()
```



Gambar 11. *Heatmap* dataset jalan aspal.

Pada gambar *heatmap* terlihat bahwa semua fitur dataset memberikan korelasi cukup rendah (kurang dari 0,3) untuk membedakan terhadap *class* datanya, hanya fitur *variance* yang sedikit lebih tinggi dari fitur lainnya.

Menerapkan Model Algoritma Pada Dataset

Dengan dataset yang telah tersedia tersebut selanjutnya diterapkan menjadi model *supervised learning*. Dataset akan dibagi menjadi dua bagian yaitu data latih (*data training*) dan data test (*data testing*) dengan porsi perbandingan adalah 2/3 dari 30 data menjadi *data training* dan 1/3 dari 30 data menjadi *data testing.*

Dengan *data training* maka setiap model algoritma klasifikasi (*Naive Bayes*, *Nearest Neighbor*, *SVM*, dan *Decision Tree*) akan dilatih menggunakan data tersebut. Selanjutnya model akan dilakukan validasi akurasinya.

Akurasi model akan dihitung berdasarkan 4 aspek, yaitu :

- Akurasi terhadap *data training* (*seen data*).
- Akurasi terhadap *data testing* (*unseen data*).
- *Cross-validation*, gabungan akurasi terhadap *data training* dan *data testing*.
- Standard deviasi.

Kode program berikut adalah implementasi dataset ke model algoritma dan pengukuran akurasi model.

```
# menentukan nama-nama kolom
colnames = ["mean", "variance", "skewness", "kurtosis",
"class"]

# membaca dataset menjadi dataframe dgn pandas
dataset = pd.read_csv("tabelaspal.csv", sep="
,header=None, names=colnames)

# cek dgn print dataframe
#print(dataset.head())

# Pemisahan dataset menjadi data training dan data test
X = dataset.iloc[:,:-1]
y = dataset.iloc[:, -1].values
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=1/3, random_state=0)

# Buat model dengan classifier Naive Bayes dan fit
model dengan data training
classifier = GaussianNB()
classifier.fit(X_train, y_train) # Fit the model dgn data
training

# Prediksi model dengan data testing
y_pred = classifier.predict(X_test)

# Cek akurasi model menggunakan accuracy_score
print("accuracy_score:
{}".format(accuracy_score(y_test, y_pred)))

# Cek akurasi model menggunakan cross validation
accuracies = cross_val_score(estimator=classifier,
X=X_train, y=y_train, cv=9)
print("Akurasi cross-val: {:.2f}
%".format(accuracies.mean()*100))
print("Standard Deviasi: {:.2f}
%".format(accuracies.std()*100))

# Hitung akurasi dari model
akurasi = classifier.score(X_train, y_train)    # Cek
akurasi dengan X_train & y_train
print("Akurasi dari model adalah : {}".format(akurasi))
```

Untuk model algoritma lain akan menggunakan basis kode program yang sama hanya akan diganti *library* model algoritmanya.

Dari kode program diatas akan dihasilkan pengukuran akurasi dan standard deviasi untuk masing-masing algoritma model seperti pada tabel 2 berikut.

Tabel. Hasil akurasi model

Algoritma model	Akurasi (data training)	Akurasi (data testing)	Akurasi cross-validation	Standard Deviasi
Decision Tree	100%	50%	74,07%	33,44%
K-Nearest Neigbor (k=3)	95%	60%	68,52%	40,40%
Support Vector Machine (SVM)	50%	40%	55,56%	17,57%
Naive Bayes (GaussianNB)	65%	30%	53,70%	34,05%

Tabel Hasil akurasi model membandingkan setiap pengukuran akurasi model dan standard deviasinya untuk masing-masing model algoritma.

4. SIMPULAN

Berdasarkan hasil implementasi dan pengujian yang dilakukan terhadap dataset ekstraksi citra momen histogram jalan aspal maka dapat disimpulkan bahwa :

1. Jika diinginkan untuk membuat model *machine learning* dari dataset tersebut untuk memprediksi klasifikasi citra jalan aspal maka model yang dapat digunakan adalah model algoritma *Decision Tree* karena memiliki akurasi *cross-validation* tertinggi.
2. Untuk keseluruhan perbandingan algoritma model klasifikasi dengan akurasi tertinggi hanya memiliki akurasi *cross-validation* 74,07%, tidak mencapai diatas 90%. Hal ini juga sesuai dengan hasil *heatmap* korelasi fitur terhadap class data yang cukup rendah.
3. Dengan akurasi *cross-validation* dari 4 algoritma model klasifikasi tersebut dengan pencapaian tertinggi hanya 74,07% maka model *machine learning* yang dibuat cukup *feasible* (handal)..

Saran

1. Dilakukan penelitian lebih lanjut dengan menambah fitur ekstraksi citra dari hanya empat fitur momen histogram tersebut agar akurasi model bisa didapatkan lebih tinggi lagi.
2. Untuk menambah fitur dataset dari citra dapat dilakukan ekstraksi citra warna (3 *channel*), dan atau memproses citra dengan teknik HOG (*Histogram of oriented gradients*).
3. Membuat aplikasi yang dapat diterapkan untuk otomasi pendeteksi jalan aspal yang memberikan informasi klasifikasi jalan rusak atau jalan tidak rusak.
4. Dapat diterapkan konversi atau pengurangan fitur dataset dengan PCA (Principal Component Analysus) untuk mendapatkan fitur dengan kovarian yang paling berpengaruh (tertinggi).

5. UCAPAN TERIMAKASIH

Peneliti pada kesempatan ini mengucapkan terimakasih kepada mahasiswi Nina Lestari yang telah membantu dalam pengambilan sampel foto (citra) jalan aspal.

6. DAFTAR PUSTAKA

[1] Defri Kurniawan, Catur Supriyanto, 2013, Optimasi Algoritma Support Vector Machine (SVM) Menggunakan Adaboost Untuk Penilaian Risiko Kredit, Jurnal Teknologi Informasi, Volume 9 Nomor 1, April 2013, pp. 38-49.

[2] Dewi Sartika, Dana Indra Sensuse, 2017, "Perbandingan Algoritma Klasifikasi Naive Bayes, Nearest Neighbour, dan Decision Tree pada Studi Kasus Pengambilan Keputusan Pemilihan Pola Pakaian, Jatisi, Vol. 1 No. 2 Maret 2017, pp. 151-161.

[3] Dini, T. (2015). Konsep dasar python. *Konsep Dasar Python*, 1–6,

[4] Gurubelajarku.com. (2020). Statistik Deskriptif. Retrieved from https://gurubelajarku.com/statistik-deskriptif/#:~:text=Histogram%20merupakan%20grafik%20dari%20distribusi,dan%20sumbu%20Y%20sebagai%20ordinat.

[5] I. Yoo, P. Alafaireet, M. Marinov, K. Pena-Hernandez, R. Gopidi, J.-F. Chang, and L. Hua, 2012, "Data Mining in Healthcare and Biomedicine: A Survey of The Literature," pp. 2431–2448

[6] J. Iawe. Han, M. Kamber, and J. Pei. (2012). Data Mining Concept and Techniques.

[7] Library binus. (2011). Computer vision, 7–8.

[8] M. Karim and R. M.Rahman. (2013). "Decision Tree and Naïve Bayes Algorithm for Classification and Generation of Actionable Knowledge for Direct Marketing," J. Softw. Eng. Appl., Vol. 6, pp. 196–206

[9] N. Jayanti, S. Puspitodjati, and T. Elida, 2008, "Teknik Klasifikasi Pohon Keputusan untuk Memprediksi Kebangkrutan Bank Berdasarkan Ratio Keuangan Bank," pp. 101–107.

[10] R. Entezari-Maleki, A. Rezaei, and B. Minaei-Bidgoli, "Comparison of Classification Methods Based on The Type of Attributes and Sample Size."

[11] rizafennisya. (2017). Pengolahan Citra Digital. Retrieved from https://rizafennisya.wordpress.com/2017/01/19/definisi-pengolahan-citra-digital/.

[12] S. H. Babic, P. Kokol, V. Podgorelec, M. Zorman, M. Sprogar, and M. M. Stiglic, 2000, "The Art of Building Decision Trees," J. Med. Syst., Vol. 24, No. 1, pp. 43–52.

[13] V. Mandasari and B. A. Tama, 2011, "Analisis Kepuasan Konsumen Terhadap Restoran Cepat Saji Melalui Pendekatan Data Mining: Studi Kasus XYZ," J. Generic, Vol. 6, No. 1, pp. 25–28.

[14] Wikipedia.org. (2020). Visi komputer. Retrieved from https://en.wikipedia.org/wiki/Computer_vision