

MACHINE LEARNING MULTI KLASIFIKASI CITRA DIGITAL

Andi Lukman¹⁾, Marwana²⁾

^{1), 2)} Teknik Informatika STIMED Nusa Palapa
Jl. Urip Sumoharjo No. 20 Gedung Graha Pena Lt. 10 Makassar
Email : uke@stimednp.ac.id¹⁾, wanastimed@gmail.com²⁾

Abstrak

Penelitian ini bertujuan membangun aplikasi *machine learning* multi klasifikasi citra digital sebagai alat bantu bagi para peneliti citra untuk mendapatkan algoritma *learning* yang optimal dalam mengenali multi kelas berbagai objek citra digital. Penelitian ini ingin menyempurnakan *machine learning* klasifikasi citra yang telah penulis teliti sebelumnya, dimana memiliki kekurangan antara lain: hanya dapat digunakan untuk binary klasifikasi, hanya dapat menginput tipe file PNG8 warna dan memerlukan waktu yang lama untuk menginput data citra digital. Algoritma yang digunakan dalam penelitian ini adalah algoritma klasifikasi *machine learning* WEKA yaitu Support Vector Machine, K-Nearest Neighbor, Naive Bayes, C4.5 Decision Tree, Logistic Regression dan Random Forest. Data set yang digunakan diambil dari California Institute of Technology bernama Caltech 101. Metode Desain sistem menggunakan diagram Unified Modeling Language. Aplikasi dibangun menggunakan bahasa pemrograman Java yang didukung oleh library *machine learning* WEKA. Black box testing digunakan untuk menguji validasi setiap fungsi pada aplikasi. Proses *learning* dan testing pengenalan citra juga akan diuji dengan menghitung persentase keberhasilan mengenali citra digital. Penelitian ini diharapkan dapat bermanfaat bagi para peneliti citra di berbagai bidang, dimana mereka dapat menggunakan *machine learning* yang dibangun untuk bereksperimen dalam menentukan algoritma *learning* yang tepat sesuai objek penelitiannya dan menggunakan algoritma tersebut untuk mengenali citra digital.

Kata kunci : Machine Learning, Multi Klasifikasi, WEKA, Support Vector Machine, K-Nearest Neighbor, Naive Bayes, C4.5 Decision Tree, Logistic Regression dan Random Forest.

1. Pendahuluan

Penelitian klasifikasi untuk pengenalan citra sangat dibutuhkan di berbagai bidang, seperti: informatika, kedokteran, kelautan, pertanian dan bisnis. Penelitian-penelitian yang telah dilakukan misalnya: klasifikasi buku [1] dan klasifikasi Breast Tissue [2] pada umumnya membuat aplikasi khusus untuk klasifikasi citra digital tertentu menggunakan algoritma *machine learning*

tertentu dengan nilai-nilai parameter optimal sesuai hasil penelitiannya. Aplikasi-aplikasi khusus tersebut hanya efektif digunakan untuk citra yang telah diteliti sebagai objek penelitiannya, sehingga setiap peneliti citra harus membangun aplikasi baru untuk objek penelitian baru. Hal ini membutuhkan keahlian dalam pemrograman, sementara para peneliti citra berasal dari berbagai bidang ilmu yang bukan programmer.

Untuk mengatasi masalah tersebut, penulis telah melakukan penelitian tentang *machine learning* klasifikasi citra digital berbasis *software as a service* (SaaS) [3]. Aplikasi yang telah dibangun memberikan fleksibilitas dalam memilih algoritma *machine learning* yang optimal dan dapat diakses dengan hanya bermodalkan browser internet. Namun aplikasi tersebut hanya dapat melakukan binary klasifikasi, dimana setiap objek citra yang akan diteliti hanya mempunyai kelas ya dan tidak, sehingga tidak dapat digunakan untuk multi klasifikasi. Aplikasi tersebut juga terbatas hanya dapat menggunakan data set citra digital bertipe PNG8 warna, sehingga data input yang tidak bertipe PNG8 warna harus dikonversi ke tipe file tersebut. Selain itu, karena aplikasi tersebut berbasis SaaS, sehingga setiap data input citra digital yang ingin digunakan harus diunggah ke server sebelum diolah, hal ini membutuhkan waktu yang lama, sementara semakin banyak data citra yang digunakan, semakin akurat hasil belajar sebuah *machine learning*.

Dari latar belakang masalah tersebut, penulis mencoba menyempurnakan hasil penelitian sebelumnya dengan melengkapinya dengan fasilitas klasifikasi multi kelas, data input yang dapat membaca beberapa jenis file citra digital dan aplikasi berbasis desktop. Penelitian ini mencoba membangun sebuah aplikasi multi klasifikasi citra digital menggunakan algoritma-algoritma *machine learning* WEKA. Hasil penelitian ini diharapkan dapat membantu para peneliti pengenalan citra tanpa harus membangun sendiri aplikasi setiap melakukan penelitian, dilengkapi dengan fleksibilitas pemilihan multi klasifikasi, algoritma *machine learning* dan jenis data citra masukan serta kecepatan dalam memasukkan data citra digital.

Beberapa penelitian yang telah dilakukan menjadi ide dasar dalam penelitian ini. Implementasi Pengolahan Citra

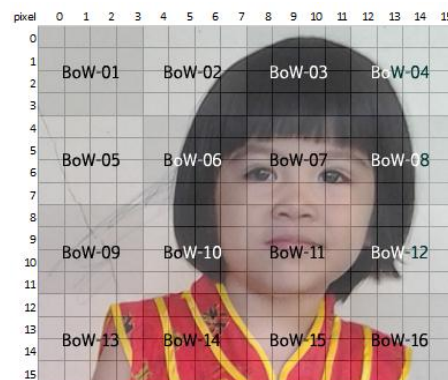
dan Algoritma LVQ untuk Pengenalan Pola Buku. Penelitian ini menggunakan Algoritma *Learning Vector Quantization* (LVQ) untuk klasifikasi buku sesuai sampulnya. Sebelum diolah menggunakan LVQ, citra terlebih dahulu dinormalisasi dan dikonversi menjadi citra biner [1]. *Breast Tissue Classification Using Gabor Filter, PCA and Support Vector Machine*. Penelitian ini menggunakan algoritma *Support Vector Machine* (SVM) untuk klasifikasi citra mamografi dalam mengenali kanker payudara. Sebelum diklasifikasi, Citra terlebih dahulu melalui proses normalisasi, ekstraksi fitur menggunakan filter gabor dan reduksi dimensi citra menggunakan PCA [2]. *Using the K-Nearest Neighbor Algorithm for the Classification of Lymph Node Metastasis in Gastric Cancer*. Citra terlebih dahulu melalui proses seleksi fitur untuk mereduksi dimesinya, kemudian digunakan *K-Nearest Neighbor* (KNN) untuk klasifikasi. Penelitian ini membuktikan kelayakan dan efektivitas dari metode *machine learning* untuk diagnosa kelenjar getah bening metastatis pada kanker lambung menggunakan data GSI [4]. *Comparing Image Classification Methods: K-Nearest-Neighbor and Support-Vector-Machines*. Penelitian ini mencoba membandingkan kemampuan klasifikasi citra metode KNN dan SVM menggunakan model *Bag of Word* (BoW). Dengan menggunakan model BoW, SVM lebih unggul dari KNN [5]. Analisis Komparasi Algoritma Klasifikasi Data Mining Untuk Prediksi Mahasiswa Non Aktif. Penelitian ini membandingkan 4 algoritma data mining yaitu *logistic regression*, *decision tree*, *naïve bayes* dan *neural network* untuk mendapatkan algoritma yang paling akurat dalam memprediksi mahasiswa non-aktif pada perguruan tinggi. Data set yang digunakan sebanyak 3861 mahasiswa Universitas Dian Nuswantoro terdiri dari data demografi dan akademik [6]. Seleksi Fitur Menggunakan *Random Forest* Dan *Neural Network*. Penelitian ini menggabungkan algoritma *Random Forest* dan *Neural Network* untuk kebutuhan klasifikasi, agar data set yang mempunyai banyak fitur. Percobaan menggunakan seleksi fitur dengan menggunakan data set iris, lung cancer dan semeion handwriting digital [7]. *Analysis of Machine Learning Algorithms using WEKA*. Penelitian ini mengimplementasikan aplikasi WEKA untuk klasifikasi dan membandingkan performa waktu learning dan ketepatan klasifikasi beberapa Algoritma *Machine Learning* [8]. *Software as a Service Untuk Machine Learning Klasifikasi Citra Digital*. Penelitian ini menawarkan fleksibilitas dalam pemilihan algoritma learning dan kemudahan akses karena menggunakan teknologi SaaS serta menggunakan algoritma-algoritma *machine learning* WEKA[3].

Citra digital merupakan gambar pada bidang dua dimensi yang direkam oleh mesin digital dan disimpan dalam bentuk file dengan format tertentu. Ukuran citra digital dinyatakan dalam pixel (*picture element*) berupa matriks dua dimensi $M \times N$ yang membentuk sebuah fungsi $f(x,y)$. Pengolahan citra merupakan proses memanipulasi dan menganalisis citra dengan bantuan komputer, dalam hal

ini mengolah informasi yang terdapat pada suatu citra digital untuk keperluan pengenalan objek secara otomatis. Ada berbagai teknik pengolahan citra tergantung kebutuhan dan keluaran yang diinginkan. Berikut beberapa teknik yang dapat digunakan yaitu: normalisasi, *Bag of Words* dan *grayscale*.

Normalisasi merupakan proses menyeragamkan ukuran citra digital inputan menjadi matriks ukuran $M \times N$. Hal ini dilakukan karena setiap citra yang diolah belum tentu mempunyai ukuran yang sama. Normalisasi juga digunakan untuk memperkecil citra digital agar jumlah pixel yang akan diolah tidak terlalu banyak. Semakin banyak jumlah pixel, semakin banyak *data set* yang menyebabkan semakin lama waktu komputasi.

Bag of Words (BoW) atau biasa juga disebut *bag of feature* merupakan teknik yang diadopsi dari *text mining* untuk kebutuhan memperkecil data set dengan mempertahankan posisi setiap bagiannya. Sebagai ilustrasi, misalnya gambar 1 merupakan citra berukuran 16X16 pixel atau 256 pixel dibagi menjadi 16 BoW. Setiap bagian terdiri dari 16 pixel (4X4 pixel).



Gambar 1. Ilustrasi *Bag of Words* 16 bagian

Grayscale adalah teknik yang digunakan untuk mengubah citra berwarna menjadi bentuk *grayscale*. Setiap pixel pada citra berwarna terbagi atas tiga bagian yaitu R (*red*), G (*green*) dan B (*blue*). Aturan yang digunakan sebagai berikut [2]:

$$(x,y,f(x,y)) \rightarrow (x,y,I(x,y))$$

$$I(i,j) = \frac{R(i,j) + G(i,j) + B(i,j)}{3} \dots\dots\dots(1)$$

Keterangan :

$f(x,y)$ = nilai intensitas lama

$I(x,y)$ = nilai intensitas *pixel grayscale*

Machine learning mempelajari bagaimana sebuah mesin atau komputer dapat belajar dari pengalaman atau bagaimana cara memprogram mesin untuk dapat belajar. *Machine learning* membutuhkan data untuk belajar sehingga biasa juga diistilahkan dengan *learn from data*

[9]. Secara garis besar ada 3 jenis metode belajar yang digunakan yaitu: *Supervised Learning*, *Unsupervised Learning* dan *Reinforcement Learning*. Khusus untuk kebutuhan klasifikasi, metode *Supervised Learning* yang sering digunakan dimana program diberikan beberapa contoh data yang telah diketahui jenis/ klasifikasinya sebagai bahan pembelajaran atau pelatihan [10].

Terdapat beberapa aplikasi *machine learning* yang telah dikembangkan oleh universitas-universitas ternama di dunia. Salah satu yang populer adalah *machine learning* WEKA. WEKA (*Waikato Environment for Knowledge Analysis*) merupakan perangkat lunak *Data Mining* yang memiliki sekumpulan algoritma standar *Machine Learning* untuk kebutuhan praproses, klasifikasi, pengelompokan, regresi, *Association Rules Mining* (ARM) dan visualisasi [8]. WEKA menyediakan *Library* pada kelas *weka.classifiers* yang dapat langsung digunakan dalam pemrograman Java [13]. Setiap data yang akan diolah menggunakan *Machine Learning* WEKA, harus memenuhi standar *dataset* dari kelas *weka.core.Instance*. Setiap instan memiliki beberapa atribut. Domain dari atribut dapat berupa Nominal (Misalnya : gunting, lampu dan cangkir), Numerik (bilangan bulat dan pecahan), String, Date (tanggal) dan Relasional. Dataset tersebut biasanya disimpan dalam format file ARFF (*Attribute-Relation File Format*) yang terdiri dari dua bagian yaitu header (menjelaskan tipe atribut) dan bagian data (meliputi data yang dipisah dengan koma). Beberapa algoritma *machine learning* yang terdapat pada kelas *weka.classifiers* tersebut antara lain : *Support Vector Machine* (functions.SMO), *K-Nearest Neighbor* (lazy.IBk), *Naive Bayes* (bayes.NaiveBayes), *C4.5 Decision Tree* (trees.J48), *Logistic Regression* (functions.Logistic) dan *Random Forest* (trees.RandomForest).

Penelitian ini menggunakan instrumen meliputi seperangkat komputer berbasis core i3 dan Perangkat lunak yaitu : Java2 SDK, Eclipse IDE for Java Developer dan *Library machine learning* WEKA.

Teknik pengumpulan data yang digunakan adalah Teknik Kepustakaan dengan mencari bahan-bahan yang berkaitan dengan materi penelitian, baik dari buku, jurnal dan internet. Data citra yang digunakan untuk data set latih, uji dan pengenalan citra adalah *Caltech 101* yaitu kumpulan objek citra digital berasal dari *California Institute of Technology* bagian penelitian *computational vision* yang diunduh dari http://www.vision.caltech.edu/Image_Datasets/Caltech101/.

Kesuksesan fungsi dari aplikasi akan diuji menggunakan metode *black box testing* sehingga dapat diketahui apakah aplikasi sukses berjalan sesuai yang diharapkan. Pengujian lainnya adalah persentase keberhasilan aplikasi dalam mengenali citra digital sesuai data set pelatihan dan

pengujian yang diberikan. Untuk menghitung persentase tersebut digunakan rumus berikut:

$$\text{Keberhasilan} = \frac{\text{Jumlah_Citra_Uji_Dikenali}}{\text{Jumlah_Citra_Uji}} \times 100\% \dots (2)$$

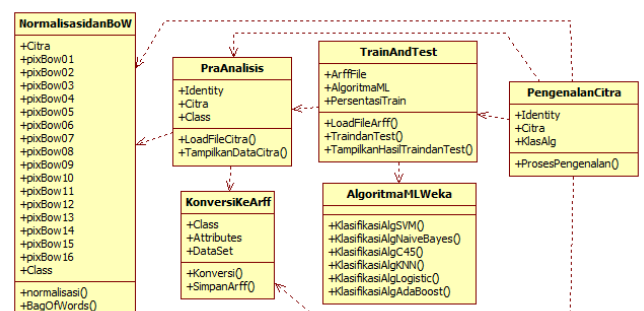
2. Pembahasan

Pembahasan terdiri dari diagram kelas, diagram aktivitas, praproses data set, pelatihan, pengujian dan pengenalan citra digital. Adapun dataset citra yang digunakan adalah citra benda yang terdiri dari cangkir, gunting dan lampu. Untuk kebutuhan pelatihan dan pengujian, masing-masing benda terdiri dari 20 citra atau secara keseluruhan 60 citra. Untuk kebutuhan pengenalan digunakan citra berbeda masing-masing 5 citra atau secara keseluruhan 20 citra. Contoh dataset tersebut dapat dilihat pada Gambar 2.



Gambar 2. Dataset Citra Digital

Kelas terdiri dari atribut dan fungsi pada suatu sistem. Diagram kelas (*class diagram*) merupakan salah satu diagram UML yang berfungsi menggambarkan struktur dan deskripsi kelas dan paket beserta hubungan satu sama lain.

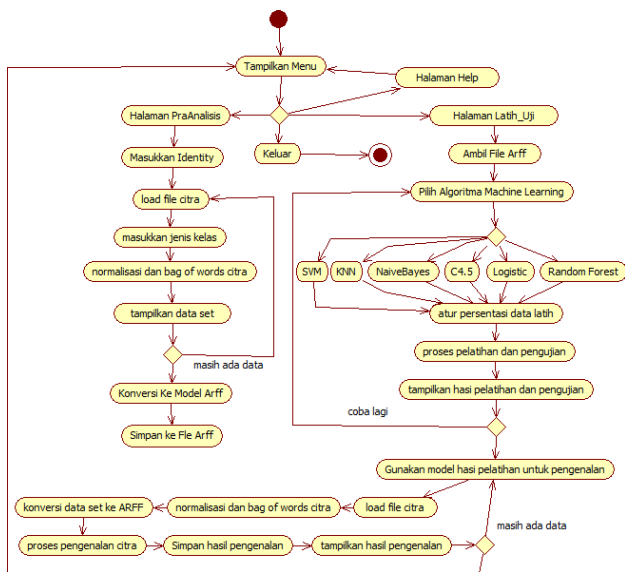


Gambar 3. Diagram Kelas aplikasi

Gambar 3 merupakan diagram kelas sistem yang dibangun. Terdapat 6 kelas diagram tersebut yaitu kelas PraAnalisis, NormalisasidanBoW, KonversiKeArff, TrainAndTest, AlgoritmaMLWeka dan PengenalanCitra. Sebelum Citra dapat diklasifikasi, citra terlebih dahulu dinormalisasi menjadi 64x64 pixel kemudian dilakukan proses *Bag Of Words* 16 bagian, setelah itu dikonversi dan disimpan ke file model ARFF agar dapat dilakukan proses

pelatihan dan pengujian menggunakan algoritma-algoritma *Machine Learning* WEKA. Pengenalan citra digital dapat dilakukan dengan menggunakan algoritma dengan persentasi terbaik sesuai hasil pelatihan dan pengujian. Diagram kelas belum menjelaskan alur aktivitas dari aplikasi. Untuk menjelaskan hal tersebut dibutuhkan diagram aktivitas.

Diagram aktivitas (*activity diagram*) merupakan salah satu diagram UML yang menggambarkan *workflow* (aliran kerja) atau aktivitas secara keseluruhan dari sistem atau proses bisnis sistem dan bukan apa yang dilakukan aktor. Diagram aktivitas mendukung perilaku paralel pada sistem. Gambar 4 merupakan diagram aktivitas dari aplikasi yang dibangun.

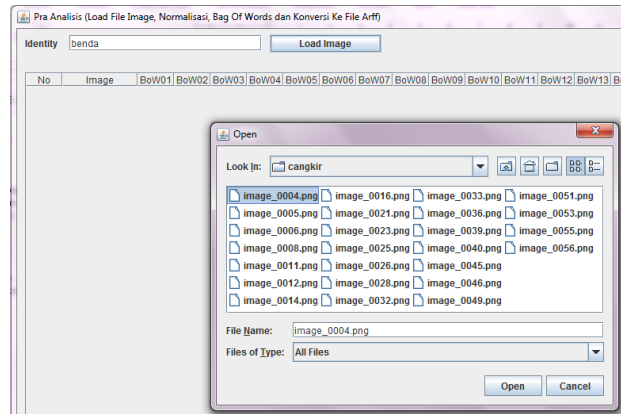


Gambar 4. Diagram Aktivitas Aplikasi

Aplikasi terdiri dari 4 menu yaitu halaman pra-analisis, halaman latih_uji, halaman help dan keluar. Pada halaman pra-analisis dilakukan proses awal sebelum citra dapat diklasifikasi, halaman latih_uji digunakan untuk melatih dan menguji algoritma *machine learning*, halaman help menampilkan cara penggunaan aplikasi dan keluar untuk keluar dari aplikasi.

Sebelum dilakukan proses pelatihan, pengujian dan pengenalan citra, terlebih dahulu citra digital melalui tahap pra-proses/pa-analisis agar dapat menjadi data set yang dapat diolah oleh algoritma-algoritma klasifikasi WEKA.

Identity dimasukkan sebagai identitas dari citra yang akan diklasifikasi. Citra digital diambil dari komputer dan melakukan proses selanjutnya. Tampilan Identity dan Ambil Citra Digital dapat dilihat pada gambar 5. Kemudian Citra digital melalui proses normalisasi berupa proses transformasi yaitu mengubah ukuran citra digital menjadi 64X64 pixel. Hal ini dilakukan agar setiap citra menjadi seragam sehingga lebih memudahkan bagi algoritma untuk mempelajarinya.



Gambar 5. Tampilan input Identity dan ambil file citra digital

Citra yang telah dinormalisasi dibagi menjadi 16 bagian dengan mempertahankan posisi setiap pixel. Hal ini dikenal dengan proses *bag of words*. Tahap pertama, mengambil nilai setiap pixel pada citra digital. Setiap pixel tersebut terdiri dari nilai R, G dan B, sehingga perlu dihitung rata-ratanya, mirip dengan proses *grayscale* menggunakan rumus (1) kemudian disimpan dalam matriks (*array*) 2 dimensi untuk mempertahankan posisi setiap pixel. Sehingga didapatkan matriks 64X64 yaitu dari (0,0) sampai (63,63) yang berisi nilai antara 0 – 255. Tahap kedua membagi matriks tersebut menjadi 16 bagian sehingga setiap bagian terdiri dari 16X16 pixel dalam bentuk matriks dengan tetap mempertahankan posisinya. Dalam mendapatkan nilai *bag of words*, dibutuhkan 16 matriks yaitu bow01 sampai bow16 dan diambil menurut posisi baris matriks. Posisi baris pertama, matriks bow01 terdiri dari pixel (0,0) sampai pixel (15,15), bow02 dari (0,16) sampai (15,31), bow03 dari (0,32) sampai (15,47), bow04 dari (0,48) sampai (15,63). Posisi baris kedua, matriks bow05 terdiri dari pixel (16,0) sampai pixel (31,15), bow06 dari (16,16) sampai (31,31), bow07 dari (16,32) sampai (31,47), bow08 dari (16,48) sampai (31,63). Posisi baris ketiga, matriks bow09 terdiri dari pixel (32,0) sampai pixel (47,15), bow10 dari (32,16) sampai (47,31), bow11 dari (32,32) sampai (47,47), bow12 dari (32,48) sampai (47,63). Posisi baris keempat, matriks bow13 terdiri dari pixel (48,0) sampai pixel (63,15), bow14 dari (48,16) sampai (63,31), bow15 dari (48,32) sampai (63,47), bow16 dari (48,48) sampai (63,63). Tahap ketiga, menghitung nilai rata-rata pixel dalam setiap bagian matriks dari matriks bow01 sampai bow16 kemudian disimpan kedalam *field* BoW01 sampai BoW16 di *datastore* bersama *field* relasi dan *field* target klasifikasinya. Namun data set tersebut belum bisa diolah oleh algoritma *machine learning* WEKA karena harus dikonversi dahulu ke format ARFF. Tampilan halaman pra-analisis dapat dilihat pada Gambar 6.

Konversi data set ke ARFF dibutuhkan agar data set dapat dikenali oleh algoritma-algoritma *Machine Learning* WEKA. Tahap Pertama, membuat bagian *header* ARFF. Setiap kelas didefinisikan menggunakan *library weka.core.FastVector* metode *addElement* yang berisi *yes*

dan *no. Library* dan metode tersebut digunakan juga untuk mengisi setiap atribut menggunakan *library weka.core.Attribute* yaitu BoW01 sampai BoW16 dan Kelas. Kemudian memasukkan *header* data set menggunakan *library weka.core.Instances*. Tahap kedua, membuat bagian data dengan cara mengambil data set yang telah tersimpan pada tabel sesuai nama relasi yang dimasukkan dan dibaca oleh komputer satu persatu dan dimasukkan kedalam data set menggunakan *library weka.core.Instance*. Proses tersebut mengalami perulangan (*loop*) sampai data habis terbaca.

No	Image	BoW01	BoW02	BoW03	BoW04	BoW05	BoW06	BoW07	BoW08	BoW09	BoW10	BoW11	BoW12	BoW13	BoW14	BoW15	BoW16	Class
53		255.0	255.0	250.0	249.5	248.8	238.1	229.7	245.6	255.0	255.0	244.5	255.0	255.0	255.0	255.0	255.0	gunting
54		143.6	89.58	65.42	56.17	185.8	148.1	48.90	48.69	186.5	153.1	49.37	48.09	123.1	77.83	58.67	140.0	lampu
55		105.3	124.4	140.6	145.5	134.5	150.1	138.3	155.8	114.2	103.6	103.0	148.7	109.5	133.2	113.7	105.1	cangkir
56		245.7	245.7	188.5	244.6	245.9	245.3	105.9	243.9	245.7	225.9	175.2	157.1	220.6	188.5	220.5	213.4	gunting
57		34.17	89.73	15.49	10.2	186.4	176.4	73.54	120.1	154.6	138.5	58.21	105.3	15.43	54.64	11.78	70.38	lampu
58		58.15	80.46	90.38	65.22	127.2	169.0	177.3	120.6	130.4	174.2	135.1	114.9	31.52	60.44	46.09	140.0	cangkir
59		252.2	252.1	216.1	189.7	220.6	219.1	178.1	175.4	168.7	199.3	166.4	175.1	251.8	252.0	215.2	185.7	gunting
60		160.8	162.0	179.9	105.9	107.1	142.2	127.2	69.74	114.5	160.4	150.0	95.75	101.1	97.44	144.5	97.84	lampu

Gambar 6. Tampilan Halaman Pra Analisis

Setelah bagian *header* dan data telah diisi lengkap, data tersebut disimpan ke dalam bentuk file ARFF. data set telah siap digunakan oleh algoritma-algoritma klasifikasi WEKA untuk kebutuhan pelatihan, pengujian dan pengenalan citra digital. Tampilan file ARFF dapat dilihat pada Gambar 7.

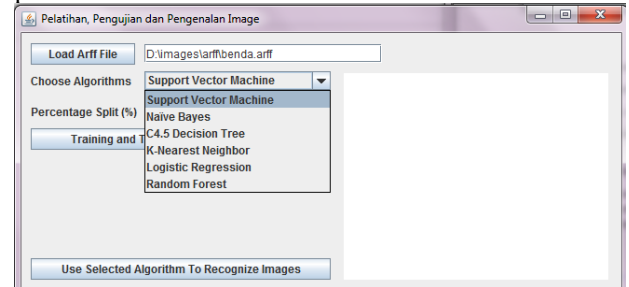
```

@relation benda
@attribute BoW01 BoW02 BoW03 BoW04 BoW05 BoW06 BoW07 BoW08 BoW09 BoW10 BoW11 BoW12 BoW13 BoW14 BoW15 BoW16
@attribute Class
@data
53 255.0 255.0 250.0 249.5 248.8 238.1 229.7 245.6 255.0 255.0 244.5 255.0 255.0 255.0 255.0 255.0 255.0 gunting
54 143.6 89.58 65.42 56.17 185.8 148.1 48.90 48.69 186.5 153.1 49.37 48.09 123.1 77.83 58.67 140.0 lampu
55 105.3 124.4 140.6 145.5 134.5 150.1 138.3 155.8 114.2 103.6 103.0 148.7 109.5 133.2 113.7 105.1 cangkir
56 245.7 245.7 188.5 244.6 245.9 245.3 105.9 243.9 245.7 225.9 175.2 157.1 220.6 188.5 220.5 213.4 gunting
57 34.17 89.73 15.49 10.2 186.4 176.4 73.54 120.1 154.6 138.5 58.21 105.3 15.43 54.64 11.78 70.38 lampu
58 58.15 80.46 90.38 65.22 127.2 169.0 177.3 120.6 130.4 174.2 135.1 114.9 31.52 60.44 46.09 140.0 cangkir
59 252.2 252.1 216.1 189.7 220.6 219.1 178.1 175.4 168.7 199.3 166.4 175.1 251.8 252.0 215.2 185.7 gunting
60 160.8 162.0 179.9 105.9 107.1 142.2 127.2 69.74 114.5 160.4 150.0 95.75 101.1 97.44 144.5 97.84 lampu
  
```

Gambar 7. Tampilan File ARFF

Sebelum melatih dan menguji algoritma *machine learning*, terlebih dahulu user mengambil file Arff kemudian memilih algoritma yang akan digunakan. Terdapat 6 pilihan algoritma yaitu : Support Vector Machine, Naïve Bayes, C4.5 Decision Tree, K-Nearest Neighbor, Logistic Regression dan Random Forest. Gambar 8 memperlihatkan pengambilan file Arff dan Pemilihan Algoritma Learning.

Setelah itu user memasukkan persentase pemisah antara pelatihan dan pengujian. Proses berikutnya adalah pelatihan dan pengujian algoritma sesuai dataset yang telah diberikan. pada proses ini aplikasi menggunakan algoritma-algoritma *Machine Learning* WEKA sesuai pilihan user.



Gambar 8. Ambil file Arff dan pemilihan Algoritma Learning

Berikut hasil pelatihan dan pengujian setiap algoritma dengan aturan persentase 70% latih dan 30% uji atau dari 60 dataset citra, 42 digunakan untuk pelatihan dan 18 digunakan untuk pengujian. Sebagai contoh, Pada Gambar 9 dengan dataset citra yang diberikan, terlihat hasil pelatihan dan pengujian tingkat kebenaran klasifikasi citra uji oleh algoritma *Support Vector Machine* yaitu 50% benar. Citra cangkir, 4 dikenali, 2 dikenali sebagai citra lampu. Citra gunting, 5 berhasil dikenali, 1 dikenali sebagai citra cangkir. Citra lampu, tidak berhasil dikenali, malah semua dikenali sebagai citra cangkir

==== Confusion Matrix ====

	a	b	c	classified as
0 0 2	a	cangkir		
0 0 0	b	gunting		
0 1 1	c	lampu		

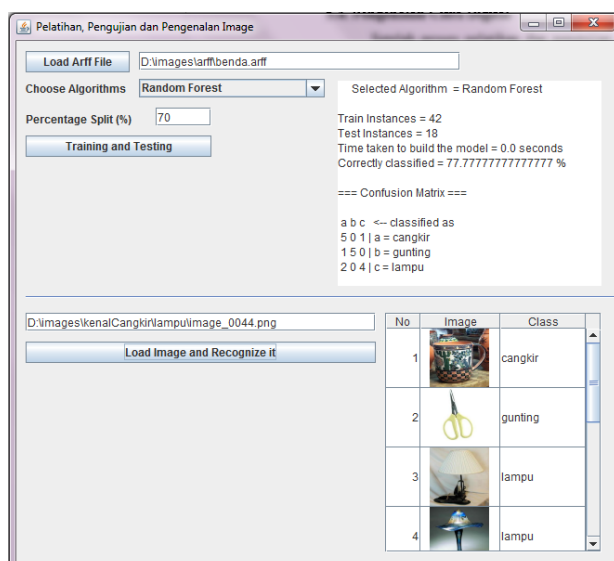
Gambar 8. Hasil Latih dan Uji Algoritma SVM

Setiap algoritma dilatih dan diuji dengan dataset dan persentasi pembagian latih uji yang sama, hasilnya adalah kemampuan klasifikasi setiap algoritma berbeda-beda. Rincian hasil tersebut dapat dilihat pada tabel 1. Kemampuan klasifikasi maksimal hanya sampai 77,77% dan minimal 33,33%.

Tabel 1. Rekapitulasi Klasifikasi Algoritma Machine Learning

Algoritma Machine Learning	Citra Uji	Jumlah dikenali Sebagai			
		Cangkir	Gunting	Lampu	Persentase
Support Vector Machine	Cangkir	4	0	2	50%
	Gunting	1	5	0	
	Lampu	6	0	0	
Naïve Bayes	Cangkir	3	0	3	44,44 %
	Gunting	0	5	1	
	Lampu	6	0	0	
C4.5 Decision Tree	Cangkir	6	0	0	50%
	Gunting	3	3	0	
	Lampu	5	1	0	
K-Nearest Neighbor	Cangkir	1	2	3	50%
	Gunting	0	5	1	
	Lampu	3	0	3	
Logistic Regression	Cangkir	3	3	1	33,33 %
	Gunting	1	3	2	
	Lampu	5	0	1	
Random Forest	Cangkir	5	0	1	77,77 %
	Gunting	1	5	0	
	Lampu	2	0	4	

Setelah proses pelatihan dan pengujian setiap algoritma learning dengan persentasi pembagian dataset yang telah diberikan, maka aplikasi telah siap digunakan untuk pengenalan citra digital. Tentunya pengenalan citra digital menggunakan algoritma *machine learning* terbaik hasil pengujian yaitu *Random Forest*. Gambar 8 memperlihatkan tampilan pengenalan citra digital dengan mengambil file citra yang diperuntukkan untuk pengenalan. Citra digital yang digunakan disini adalah citra yang tidak termasuk dalam 60 dataset keperluan pelatihan dan pengujian.

**Gambar 8.** Tampilan Pengenalan Citra Digital

3. Kesimpulan

Penelitian ini berhasil membangun *machine learning* multi klasifikasi citra digital yang fleksibel untuk kebutuhan pelatihan, pengujian dan pengenalan. Hasil pengujian keberhasilan klasifikasi dengan 60 dataset citra cangkir, gunting dan lampu menempatkan algoritma Random Forest sebagai algoritma learning terbaik yaitu 77,77% berhasil mengklasifikasi citra digital, Sehingga algoritma inilah yang digunakan dalam proses pengenalan citra digital.

Daftar Pustaka

- [1] Lukman, Andi, "Implementasi Pengolahan Citra dan Algoritma LVQ Untuk Pengenalan Pola Buku", *Seminar Nasional Informatika (SNIf)*, hal. 145-151, Okt. 19, 2012.
- [2] Hajare, Pravin S. dan Dixit, Vaibhav V, "Breast Tissue Classification Using Gabor Filter, PCA and Support Vector Machine", *International Journal of advancement in electronics and computer engineering (IJAECE)*, Vol. 1 Issue 4, hal. 116-119, Juli, 2012.
- [3] Lukman, Andi dan Marwana, "Software as a Service Untuk Machine Learning Klasifikasi Citra Digital", *Seminar Nasional Teknologi Informasi dan Multimedia (Semnasteknomedia)*, buku 2, hal. 2.01.31-2.01.36, Feb. 8, 2014
- [4] Li, Chao, dkk., "Using the K-Nearest Neighbor Algorithm for the Classification of Lymph Node Metastasis in Gastric Cancer", *Journal of Computational and Mathematical Methods in Medicine*, Vol. 2012, hal. 1-11, Okt. 2012.
- [5] Kim, Jinho, dkk., "Comparing Image Classification Methods-K-Nearest-Neighbor and Support-Vector-Machines", *Proceedings of Applied Mathematics in Electrical and Computer Engineering*, hal. 133-138, January 25-27, 2012.
- [6] Hastuti, Khafizh, "Analisis Komparasi Algoritma Klasifikasi Data Mining Untuk Prediksi Mahasiswa Non Aktif", *Proc. Semantik 2012*, hal. 241 - 249, Juni, 23, 2012.
- [7] Saputra, Wahyuni S.J, dkk., "Seleksi Fitur Menggunakan Random Forest Dan Neural Network", *Proc. IES 2011*, hal. 93-97, Oktober 26, 2011.
- [8] Desai, Aaditya, dan Rai, Sunil, "Analysis of Machine Learning Algorithms using WEKA", *Proc. ICWET 2012*, hal. 27-32, Maret, 2012.
- [9] Alpaydin, Ethem, *Introduction to Machine Learning*, Second Edition, London: MIT Press, 2010.
- [12] Harrington, Peter, *Machine Learning .in Action*, , New York: Manning, 2012.
- [13] Witten, Ian H., dkk, *Data Mining Practical Machine Learning Tools and Techniques*, Third Edition, Burlington: Morgan Kaufmann Publishers, 2011.