

SOFTWARE AS A SERVICE UNTUK MACHINE LEARNING 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

Machine Learning klasifikasi citra digital yang telah ada hanya dapat digunakan untuk kebutuhan khusus objek penelitian citra tertentu dan belum dapat digunakan secara online dan multiuser menggunakan web browser, hal ini mengurangi fleksibilitas para pengguna. Penelitian ini bertujuan menerapkan teknologi *Software as a Service* untuk *machine learning* klasifikasi citra digital agar aplikasi klasifikasi citra digital dapat digunakan oleh para peneliti citra secara multiuser dan tanpa proses instalasi aplikasi. Metode penelitian ini memanfaatkan data set citra digital Caltech 101 yang diambil dari Computational vision Institut Teknologi California. Aplikasi dibangun menggunakan bahasa pemrograman Java dimana untuk sisi client menggunakan Google Web Toolkit dan sisi server menggunakan Java Servlet yang kemudian dideploy ke Google App Engine. Algoritma yang digunakan adalah algoritma klasifikasi *Machine Learning* WEKA yaitu: *Support Vector Machine*, *K-Nearest Neighbor*, *Naïve Bayes*, *C4.5 Decision Tree*, *Logistic Regression* dan *Random Forest*. Fungsi aplikasi sebagai *Software as a Service* diuji menggunakan pengujian *black box*. Hasil penelitian memperlihatkan bahwa teknologi *Remote Procedure Call* dengan metode *Asynchronous* dapat diterapkan agar aplikasi dapat diakses secara online menggunakan web browser secara multiuser. Pengujian dilakukan oleh 4 pengguna yang secara bersamaan mengakses aplikasi pada komputer berbeda dengan fungsi-fungsi: *login/logout*, mengakses aplikasi tanpa instalasi hanya menggunakan web browser, menggunakan *pra-proses data set* dan menggunakan jenis algoritma *machine learning* yang sama dengan data set berbeda. Hasil pengujian memperlihatkan setiap fungsi sukses melewati tahap pengujian *black box*. Penelitian ini dapat disimpulkan bahwa teknologi *Software as a Service* dapat diterapkan pada *machine learning* klasifikasi citra digital.

Kata kunci: *Software as a Service*, *Machine Learning*, *Klasifikasi Citra Digital*.

1. Pendahuluan

Software as a Service (SaaS) merupakan salah satu layanan Teknologi *Cloud Computing* yang memudahkan pengguna menggunakan aplikasi secara online

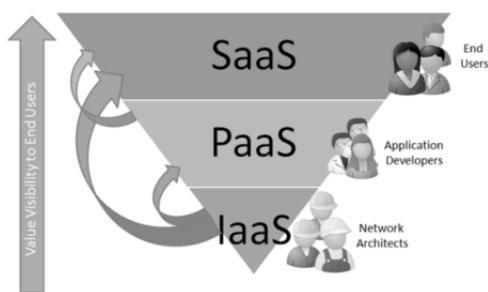
dimanapun dan kapanpun (selama ada koneksi internet) tanpa harus melakukan instalasi. Aplikasi berbasis SaaS dapat digunakan oleh beberapa pengguna dalam waktu bersamaan (*multiuser*). Penelitian-penelitian aplikasi *machine learning* klasifikasi citra digital sebelumnya masih bersifat khusus dan kurang fleksibel karena hanya digunakan untuk citra digital tertentu, aplikasi tersebut harus dibangun serta melalui proses instalasi pada komputer pengguna sebelum digunakan dan tidak dapat digunakan secara online di internet dan tidak *multiuser*.

Dengan adanya permasalahan tersebut, kami menganggap perlu melakukan penelitian berjudul "*Software as a Service Untuk Machine Learning Klasifikasi Citra Digital*". Hasil penelitian ini diharapkan dapat membantu para peneliti klasifikasi citra digital tanpa harus membangun sendiri aplikasi setiap melakukan penelitian dengan fleksibilitas dalam hal kemudahan akses aplikasi, memilih algoritma klasifikasi dan dapat digunakan untuk berbagai objek citra digital.

Penelitian ini bertujuan menerapkan teknologi SaaS untuk aplikasi *machine learning* klasifikasi citra digital agar aplikasi yang dibangun dapat digunakan oleh para peneliti citra secara *multiuser* dan tanpa proses instalasi aplikasi.

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 sampelnya. 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 dimensinya, 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 [3]. *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 [4]. 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 [5]. 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* [6]. *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 [7]. Penelitian-penelitian tersebut dapat digunakan untuk klasifikasi citra digital, namun tidak memiliki fleksibilitas dalam hal pengaksesan aplikasi. Penelitian lain yang dapat dijadikan ide untuk melengkapi hal tersebut yaitu *Software as a Service for Data Scientists*. Penelitian ini menawarkan solusi teknologi SaaS untuk manajemen data penelitian, seperti kemudahan akses data antar para peneliti atau fasilitas-fasilitas penelitian [8].

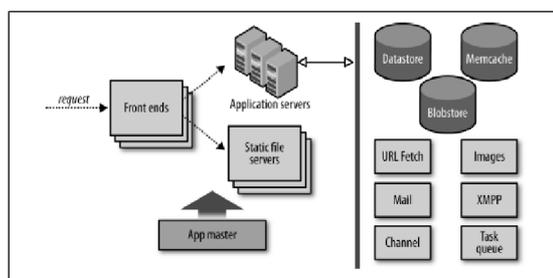


Gambar 1. Arsitektur Layer Cloud Computing

Software as a Service (SaaS) merupakan salah satu bentuk layanan *Cloud Computing*. SaaS menyediakan *service* dalam bentuk *software* yang sudah jadi dan dapat digunakan dimanapun dan kapanpun (selama terdapat akses ke internet) dengan syarat-syarat tertentu yang sudah di tentukan oleh penyedia jasa. Posisi atau *layer* SaaS dalam teknologi *Cloud Computing* [9] dapat dilihat pada Gambar 1. Pihak *developer* tidak perlu berinvestasi pada lapisan-lapisan yang ada di bawahnya untuk membangun aplikasi berbasis SaaS, cukup

menggunakan *Platform as a Service* (PaaS) yang telah tersedia di internet. Salah satu PaaS yang dapat digunakan adalah layanan *Google App Engine* (GAE) dari google.

Google App Engine (GAE) menyediakan sekumpulan API (*Application Programming Interface*) dalam bentuk SDK (*Software Development Kit*) yang bisa diunduh secara bebas. Selain itu, produk GAE sendiri juga bebas atau gratis untuk digunakan dengan batasan-batasan *resources* yang diberikan, namun dalam beberapa hal tertentu, *resources* yang tersedia sudah mencukupi. Jika membutuhkan tambahan sumber daya sesuai keinginan, *user* dapat menggunakan versi berbayar. Arsitektur GAE khususnya penanganan permintaan (*request handling*) secara umum dapat dilihat pada Gambar 2 [10]. Saat ini, untuk menggunakan GAE, anda dapat menggunakan empat macam bahasa pemrograman, yaitu JAVA, PHYTON, GO dan PHP. Khusus untuk teknologi Java yang umum dipakai di GAE adalah Servlet dan JSP.



Gambar 2. Arsitektur penanganan permintaan GAE

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* [11]. 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 [12].

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 [7]. 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 : ya dan tidak), 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 tersebut antara lain : *Support Vector Machine*, *K-Nearest Neighbor*, *Naive Bayes*, *C4.5 Decision Tree*, *Logistic Regression* dan *Random Forest*.

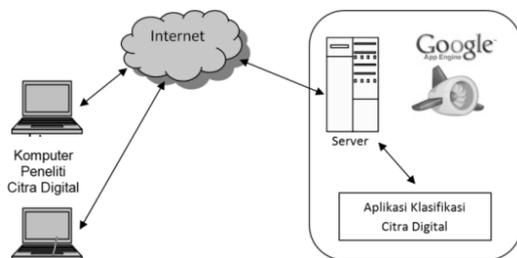
Penelitian ini menggunakan instrumen meliputi seperangkat komputer berbasis core i3 dan Perangkat lunak yaitu : Java2 SDK, Eclipse for Java EE Developer, Google Plugin for Eclipse, Google App Engine SDK, Google Web Toolkit dan Web Browser.

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 bertipe PNG 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 sebagai SaaS akan diuji menggunakan metode *black box testing* sehingga dapat diketahui apakah aplikasi sukses berjalan sesuai yang diharapkan.

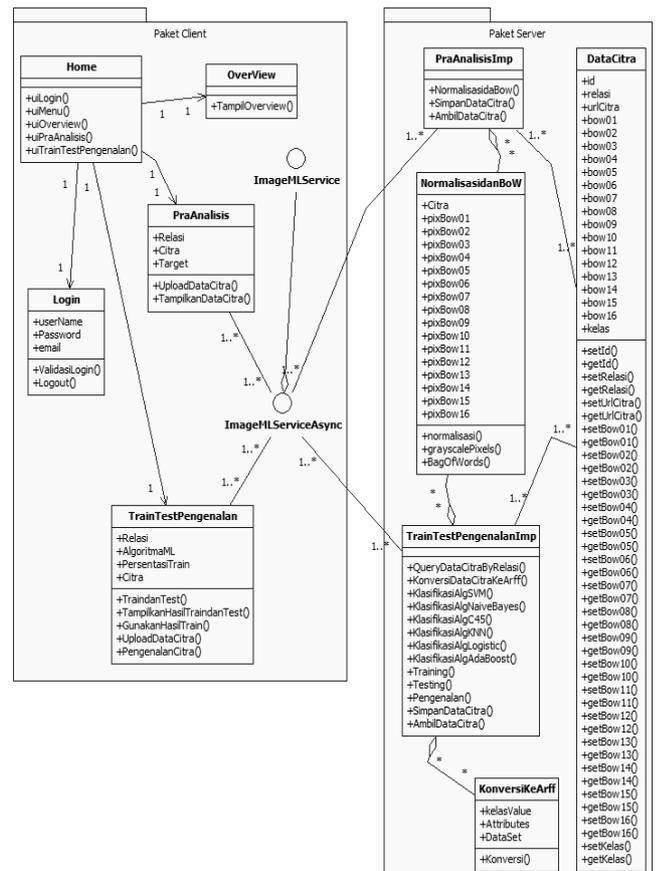
2. Pembahasan

Pada bagian ini akan dibahas hasil penelitian yang terdiri dari rancangan arsitektur SaaS, diagram kelas, diagram *deployment*, proses *multiuser SaaS* dan tampilan aplikasi. kemudian aplikasi diuji kehandalannya sebagai SaaS.



Gambar 3. Arsitektur Aplikasi menggunakan teknologi SaaS

Penelitian ini menerapkan teknologi SaaS pada aplikasi *machine learning* klasifikasi citra Digital menggunakan algoritma-algoritma *machine learning* WEKA. Aplikasi berjalan di *cloud* sehingga dapat diakses secara *multiuser*. Desain arsitektur aplikasi dapat dilihat pada Gambar 3.

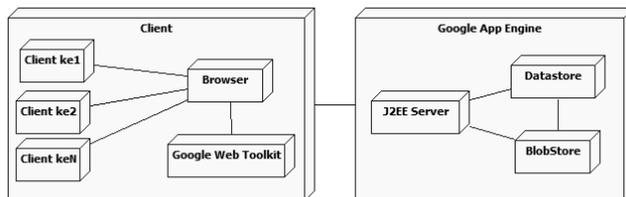


Gambar 4. Diagram kelas aplikasi

Diagram kelas (*class diagram*) berfungsi menggambarkan struktur dan deskripsi kelas dan paket beserta hubungan satu sama lain. Gambar 4 memperlihatkan diagram kelas sistem yang dibangun. Terdapat 2 paket pada diagram tersebut yaitu paket *Client* dan *Server* dimana setiap paket terdiri dari beberapa kelas. Kelas-kelas pada paket *Client* dapat langsung dijalankan pada *web browser* pengguna (peneliti citra) karena dibuat menggunakan *Google Web Toolkit* (GWT) yang mengkonversi bahasa java menjadi *java script* sehingga dapat dijalankan di *web browser*. Sementara kelas-kelas pada paket *Server* hanya dapat dijalankan pada sisi server yaitu *Servlet* pada *Google App Engine* (GAE). Komunikasi antara client dan server dijumpai oleh *interface* *ImageMLService* dan *ImageMLServiceAsync*.

Diagram *Deployment* menggambarkan konfigurasi komponen pada saat instalasi aplikasi. Diagram ini juga menunjukkan tata letak sistem secara fisik, menampilkan bagian-bagian perangkat lunak yang berjalan pada bagian-bagian perangkat keras. Diagram ini sangat tepat digunakan untuk memodelkan sistem *client/server*. Gambar 5 menunjukkan diagram *deployment* sistem yang dibangun. Beberapa client (*multiuser*) dapat mengakses aplikasi menggunakan browser, dimana aplikasi ditampilkan dalam bentuk *java script* yang telah

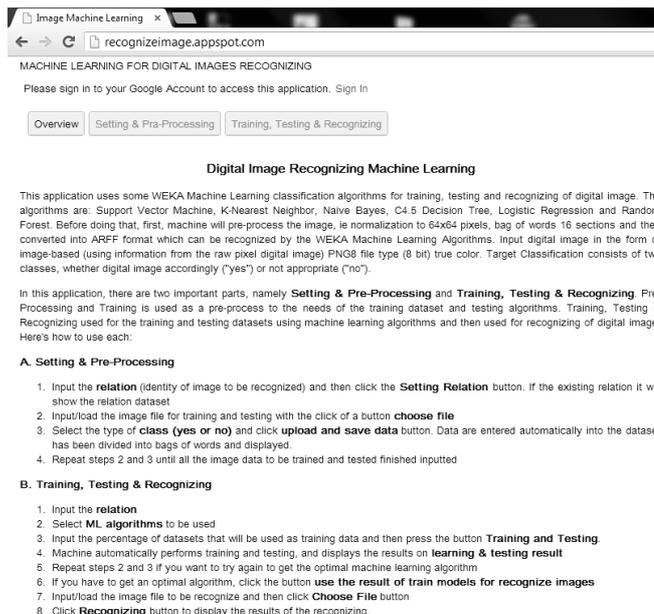
dikonversi oleh *Google Web Toolkit*. Client meminta layanan ke *server* (dalam hal ini *Google App Engine*) menggunakan teknik *Remote Procedure Call (RPC)*, kemudian *server* mengelola permintaan tersebut menggunakan *J2EE Server*. Data citra akan disimpan oleh *Blobstore*, sementara data set akan disimpan oleh *Datastore*. Setelah selesai memproses permintaan *client*, *server* akan mengembalikan hasil pengolahan data ke *client* dan ditampilkan pada *web browser*.



Gambar 5. Diagram Deployment sistem

Kehandalan aplikasi berbasis *Software as a Service (SaaS)*, salah satunya adalah kemampuan dalam melayani pengguna secara *multiuser*. Diagram class pada Gambar 4 memperlihatkan teknologi *interface* yang menjembatani antara paket *client* dan paket *server*. Paket *Client* meminta layanan (*request*) kepada Paket *Server* melalui *interface ImageMLService* dan *ImageMLServiceAsync*. Layanan permintaan seperti ini menggunakan teknologi *Remote Procedure Call (RPC)*. *RPC* menggunakan metode *Asynchronous*, dimana *client* tidak perlu menunggu sampai *server* melayaninya karena dapat menyebabkan browser *hang* jika *server* sangat sibuk. *Client* cukup mengirim permintaan ke *server* kemudian memutuskan koneksi sambil menanti kiriman hasil pengolahan data dari *server*. Jika *server* telah siap dan selesai memproses permintaan dari *client*, secara otomatis *server* mengirimkan kembali hasil pengolahan data tersebut ke *client*. Kecepatan dan kehandalan dalam memproses permintaan *client* bergantung pada *server google* yang berada dibanyak tempat dan terkoneksi di seluruh dunia. Kemampuan inilah yang membuat aplikasi *machine learning* yang dibangun dapat diakses secara bersamaan oleh banyak pengguna (*multiuser*) dengan kecepatan proses yang handal. Gambar 5 memperlihatkan aplikasi yang dibangun secara fisik menerapkan teknik *multiuser SaaS* pada *Platform as a Service (PaaS) Google App Engine*.

Aplikasi dapat diakses melalui <http://gcdc2013-image.ml.appspot.com/> dan tampilannya dapat dilihat pada Gambar 6, dimana terdiri dari *Sign in (login)*, *overview*, *setting & pra-processing* dan *training, testing & recognizing*. Halaman *Overview* menjelaskan tentang ruang lingkup aplikasi dan cara menggunakannya. Setiap pengguna wajib memiliki *Google account* untuk *sign in* agar dapat menggunakan aplikasi.



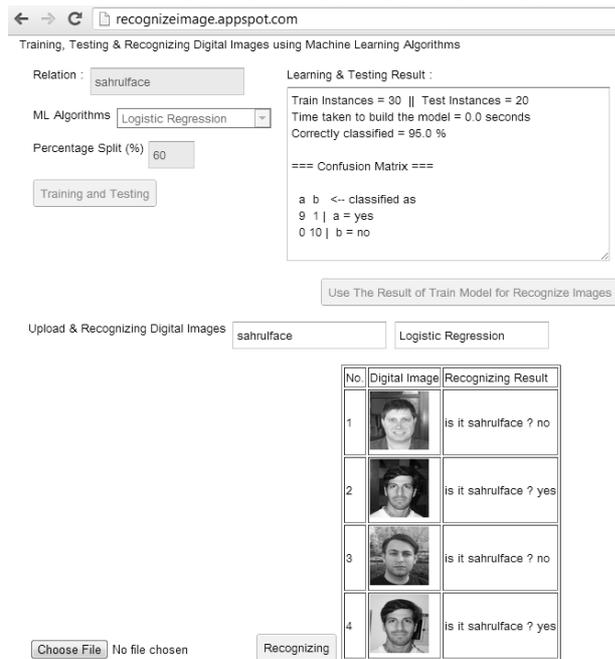
Gambar 6. Tampilan Halaman Overview

Pada halaman *setting & pra-processing*, pengguna memasukkan identitas citra digital yang akan dikenali (*relation*), kemudian mengunggah beberapa citra digital untuk bahan pelatihan dan pengujian. Setiap citra digital tersebut mengalami praproses berupa normalisasi dan *bag of words*, Tampilan halaman tersebut dapat dilihat pada Gambar 7.

No. Image	BoW 01	BoW 02	BoW 03	BoW 04	BoW 05	BoW 06	BoW 07	BoW 08	BoW 09	BoW 10	BoW
1	86.7734375	61.83984375	46.29296875	48.6171875	79.67421875	133.5234375	143.3125	44.51171875	36.7890625	163.984375	163
2	201.37890625	166.92578125	163.95703125	131.50765625	178.26171875	204.88328125	202.45703125	79.3671875	192.4046875	216.9140625	210
3	166.4609375	74.44921875	70.56640625	128.1484375	108.42578125	126.23046875	135.203125	88.01171875	170.81640625	166.90625	166
4	137.77734375	93.89359375	69.35546875	109.65625	117.11328125	151.921875	195.44921875	55.1494375	136.33984375	182.67578125	188
5	145.74609375	91.63671875	89.0859375	140.19140625	121.59375	134.1875	157.93359375	54.99609375	147.1484375	158.0234375	155
6	49.90625	64.484375	61.453125	48.921875	67.24609375	163.61328125	150.09765625	39.296875	73.12890625	176.2578125	159
7	90.640625	85.6640625	81.87890625	118.55078125	83.953125	164.31640625	40.578125	61.5390625	88.08203125	158.61328125	158
8	138.5390625	144.51171875	133.7578125	145.39453125	68.27734375	183.13671875	170.51953125	69.21484375	67.45703125	189.953125	178
9	187.0390625	104.109375	106.0	129.82421875	112.21875	158.24609375	132.37109375	126.58984375	169.515625	185.5703125	180
10	194.2265625	118.53125	115.3515625	194.5703125	152.30859375	102.66796875	152.890625	126.6328125	155.23046875	116.86328125	143
11	224.34375	127.03125	131.33984375	231.5234375	162.109375	153.921875	139.03125	166.1328125	219.62109375	172.75390625	166
12	103.7265625	107.171875	89.26171875	89.8984375	111.171875	137.40625	183.8515625	61.0078125	81.828125	207.2926875	197
13	132.44921875	72.1015625	63.6640625	119.59375	104.5625	139.56640625	167.01171875	72.8203125	122.59765625	160.55078125	161

Gambar 7. Halaman Setting & Pra-Processing

Halaman *training, testing & recognizing* digunakan untuk proses pelatihan dan pengujian dataset menggunakan algoritma-algoritma *machine learning*. Pengguna dapat memilih algoritma yang mempunyai persentasi keberhasilan tertinggi untuk digunakan dalam mengenali citra digital. Gambar 8 memperlihatkan halaman tersebut.



Gambar 8. Halaman Training, Testing & Recognizing

Pengujian *black box* untuk aplikasi sebagai SaaS dilakukan agar dapat diketahui bahwa aplikasi yang telah dibangun sukses menerapkan teknologi SaaS sesuai yang diharapkan. Pengujian ini dilakukan oleh 4 pengguna yang secara bersamaan menggunakan aplikasi pada komputer berbeda, yaitu : arkanialmahdi, wanastimed, uke@stimednp.ac.id dan info@stimednp.ac.id. Fungsi-fungsi aplikasi yang diuji yaitu beberapa pengguna dapat: login dan logout secara bersamaan; menggunakan aplikasi tanpa instalasi, hanya menggunakan web browser; menggunakan praproses data set citra secara bersamaan; menggunakan jenis algoritma *machine learning* yang sama secara bersamaan.

Tabel 1. Hasil pengujian *black box* untuk login dan logout secara multiuser

No.	Jenis Input	Output yang diharapkan	Hasil
1	4 pengguna secara bersamaan mengakses aplikasi, menekan <i>link Sign In</i> dan memasukkan <i>user</i> dan <i>password</i> akun <i>google-nya</i>	Semua pengguna berhasil <i>login</i> secara bersamaan.	Sukses
2	4 pengguna secara bersamaan menekan <i>link Sign Out</i>	Semua pengguna berhasil <i>logout</i> secara bersamaan..	Sukses

Hasil pengujian pada Tabel 1 memperlihatkan bahwa semua sub bagian yang ada pada modul login dan logout secara *multiuser* tersebut sudah sesuai dengan harapan.

Tabel 2. Hasil pengujian *black box* untuk mengakses tanpa instalasi dan hanya menggunakan web browser

No.	Jenis Input	Output yang diharapkan	Hasil
1	4 pengguna secara bersamaan mengakses aplikasi menggunakan <i>web browser</i> masing-masing tanpa instalasi aplikasi	Semua pengguna berhasil dapat mengakses aplikasi secara bersamaan.	Sukses

Hasil pengujian pada Tabel 2 memperlihatkan bahwa fungsi akses tanpa instalasi dan hanya menggunakan *web browser* sudah sesuai dengan harapan.

Tabel 3. Hasil pengujian *black box* menggunakan praproses data set citra secara bersamaan

No.	Jenis Input	Output yang diharapkan	Hasil
1	Pengguna 1 melakukan praproses data set citra wajah dalam waktu bersamaan dengan ketiga pengguna lainnya	Pengguna 1 berhasil melakukan praproses data set citra wajah	Sukses
2	Pengguna 2 melakukan praproses data set citra kupu-kupu dalam waktu bersamaan dengan ketiga pengguna lainnya	Pengguna 2 berhasil melakukan praproses data set citra kupu-kupu	Sukses
3	Pengguna 3 melakukan praproses data set citra bunga matahari dalam waktu bersamaan dengan ketiga pengguna lainnya	Pengguna 1 berhasil melakukan praproses data set bunga matahari	Sukses
4	Pengguna 4 melakukan praproses data set citra cangkir dalam waktu bersamaan dengan ketiga pengguna lainnya	Pengguna 2 berhasil melakukan praproses data set citra cangkir	Sukses

Hasil pengujian pada Tabel 3 memperlihatkan bahwa fungsi praproses data set citra dapat diakses secara bersamaan berjalan sesuai dengan harapan.

Hasil pengujian pada Tabel 4 memperlihatkan bahwa fungsi menggunakan jenis algoritma yang sama secara *multiuser* berjalan sesuai dengan harapan.

Setiap bagian dari fungsi SaaS pada aplikasi telah sukses melewati tahap pengujian *black box*. Hal tersebut menandakan bahwa setiap fungsi aplikasi sebagai SaaS sudah sesuai dengan harapan.

Tabel 4. Hasil pengujian black box menggunakan jenis algoritma yang sama secara multiuser

No.	Jenis Input	Output yang diharapkan	Hasil
1	4 pengguna secara bersamaan menggunakan algoritma SVM dengan data set berbeda	Semua pengguna berhasil menggunakan algoritma SVM.	Sukses
2	4 pengguna secara bersamaan menggunakan algoritma Naïve Bayes dengan data set berbeda	Semua pengguna berhasil menggunakan algoritma Naïve Bayes	Sukses
3	4 pengguna secara bersamaan menggunakan algoritma C4.5 dengan data set berbeda	Semua pengguna berhasil menggunakan algoritma C4.5	Sukses
4	4 pengguna secara bersamaan menggunakan algoritma KNN dengan data set berbeda	Semua pengguna berhasil menggunakan algoritma KNN	Sukses
5	4 pengguna secara bersamaan menggunakan algoritma Logistic Regression dengan data set berbeda	Semua pengguna berhasil menggunakan algoritma Logistic Regression.	Sukses
6	4 pengguna secara bersamaan menggunakan algoritma Random Forest dengan data set berbeda	Semua pengguna berhasil menggunakan algoritma Random Forest	Sukses

3. Kesimpulan

Penelitian ini berhasil menerapkan teknologi SaaS untuk aplikasi *machine learning* klasifikasi citra digital sehingga aplikasi dapat digunakan oleh para peneliti citra secara *multiuser* dan tanpa proses instalasi. Hal ini dibuktikan dengan pengujian *black box* pada setiap fungsi pemanfaatan SaaS pada aplikasi.

Daftar Pustaka

- [1] Lukman, Andi, "Implementasi Pengolahan Citra dan Algoritma LVQ Untuk Pengenalan Pola Buku", *Seminar Nasional Informatika (SNI)*, 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] 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.
- [4] 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.
- [5] Hastuti, Khafiizh, "Analisis Komparasi Algoritma Klasifikasi Data Mining Untuk Prediksi Mahasiswa Non Aktif", *Proc. Semantik 2012*, hal. 241 - 249, Juni, 23, 2012.
- [6] Saputra, Wahyuni S.J, dkk., "Seleksi Fitur Menggunakan Random Forest Dan Neural Network", *Proc. IES 2011*, hal. 93-97, Oktober 26, 2011.
- [7] Desai, Aaditya, dan Rai, Sunil, "Analysis of Machine Learning Algorithms using WEKA", *Proc. ICWET 2012*, hal. 27-32, Maret, 2012.
- [8] Allen, Bryce, dkk., "Software as a Service for Data Scientists", *Communications of The ACM*, Vol. 55, No. 2, hal. 81-88, Februari, 2012.
- [9] Furht, Borko dan Escalante, Armando, *Handbook of Cloud Computing*, New York: Springer Science+Business Media, 2010.
- [10] Sanderson, Dan, *Programming Google App Engine*, Second Edition, Sebastopol: O'Reilly Media Inc., 2013.
- [11] 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.

Biodata Penulis

Andi Lukman, S.Kom, M.T., memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Teknik Informatika STMIK Dipanegara Makassar, lulus tahun 2000. Memperoleh gelar Magister Teknik (M.T.) Program Pasca Sarjana Magister Teknik Elektro Konsentrasi Teknik Informatika Universitas Hasanuddin (UNHAS) Makassar, lulus tahun 2013. Saat ini menjadi Dosen di STIMED Nusa Palapa Makassar.

Marwana, S.Kom, M.T., memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Manajemen Informatika STMIK Dipanegara Makassar, lulus tahun 2002. Memperoleh gelar Magister Teknik (M.T.) Program Pasca Sarjana Magister Teknik Elektro Konsentrasi Teknik Informatika Universitas Hasanuddin (UNHAS) Makassar, lulus tahun 2013. Saat ini menjadi Dosen di STIMED Nusa Palapa Makassar.