

**IDENTIFIKASI LAHAN KOSONG KOTA BATU BERBASIS
CITRA *GOOGLE EARTH* MENGGUNAKAN
ALGORITMA *CONVOLUTIONAL*
NEURAL NETWORK (CNN)**

SKRIPSI

Oleh :
ISLAM NUR ALAM
NIM. 16650096



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2020**

**IDENTIFIKASI LAHAN KOSONG KOTA BATU BERBASIS
CITRA *GOOGLE EARTH* MENGGUNAKAN
ALGORITMA *CONVOLUTIONAL*
NEURAL NETWORK (CNN)**

SKRIPSI

**Diajukan kepada:
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh:
ISLAM NUR ALAM
NIM. 16650096**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2020**

LEMBAR PERSETUJUAN
IDENTIFIKASI LAHAN KOSONG KOTA BATU BERBASIS
CITRA *GOOGLE EARTH* MENGGUNAKAN
ALGORITMA *CONVOLUTIONAL*
***NEURAL NETWORK* (CNN)**

SKRIPSI

Oleh :
ISLAM NUR ALAM
NIM. 16650096

Telah Diperiksa dan Disetujui untuk Diuji

Tanggal : 17 Juni 2020

Dosen Pembimbing I

Dosen Pembimbing II

Irwan Budi Santoso, M.Kom
NIP. 19770103 201101 1 004

Fathurrochman, M.Kom
NIP. 19700731 200501 1002

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

LEMBAR PENGESAHAN

**IDENTIFIKASI LAHAN KOSONG KOTA BATU BERBASIS
CITRA *GOOGLE EARTH* MENGGUNAKAN
ALGORITMA *CONVOLUTIONAL*
NEURAL NETWORK (CNN)**

SKRIPSI

Oleh:
ISLAM NUR ALAM
NIM. 16650096

Telah Dipertahankan di Depan Dewan Penguji
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Pada Tanggal Juni 2020

Susunan Dewan Penguji

Tanda tangan

- | | | | |
|-----------------------|---|--|---------|
| 1. Penguji Utama | : | <u>A'la Syauqi, M.Kom</u>
NIP. 19771201 200801 1 007 | (.....) |
| 2. Ketua Penguji | : | <u>Dr. M. Amin Hariyadi</u>
NIP. 19670018 200501 1 001 | (.....) |
| 3. Sekretaris Penguji | : | <u>Irwan Budi Santoso, M.Kom</u>
NIP. 19770103 201101 1 004 | (.....) |
| 4. Anggota Penguji | : | <u>Fathurrochman, M.Kom</u>
NIP. 19700731 200501 1002 | (.....) |

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdiان
NIP. 19740424 200901 1 008

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama : Islam Nur Alam

NIM : 16650096

Fakultas/Jurusan : Sains dan Teknologi/Teknik Infomatika

Judul Skripsi : Identifikasi Lahan Kosong Kota Batu Berbasis Citra *Google Earth* Menggunakan Algoritma *Convolutional Neural Network* (CNN)

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 17 Juni 2020
Yang membuat pernyataan,



Islam Nur Alam
NIM.16650096

HALAMAN MOTTO

“Orang yang suka berkata jujur mendapatkan tiga hal: kepercayaan, cinta dan rasa hormat”

- Ali Bin Abi Thalib -



HALAMAN PERSEMBAHAN

Alhamdulillah puji syukur kehadiran Allah SWT yang memberikan kekuatan kepada saya hingga bisa sampai menyelesaikan kuliah S1 di kampus UIN MAULANA MALIK IBRAHIM MALANG. Sholawat serta salam kepada Nabi Muhammad SAW, yang membawa petunjuk kepada seluruh umat manusia.

Alhamdulillah, terima kasih kepada kedua orang tuas saya, yang telah membesarkan dan mendidik saya dari kecil hingga sekarang bisa menyelesaikan kuliah saya, yang tiap hari mendoakan saya tiada henti, dan selalu mendukung saya dalam keadaan apapun. Tak lupa terimakasih untuk sahabat terbaik saya Lutifah Husnul Howiyah yang selalu memberi semangat dan dukungan yang tiada henti sampai pengerjaan skripsi ini selesai.

Terima kasih kepada dosen-dosen yang telah sabar dan ikhlas dalam mendidik saya hingga mampu melewati seluruh ujian dari semua mata kuliah yang saya tempuh, terutama kepada Bapak Santoso, M.Kom sebagai dosen wali yang selalu sabar dalam membuka wawasan saya. Tidak lupa juga saya ucapkan terimakasih kepada Bapak Irwan Budi Santoso, M.Kom dan Bapak Fathurrochman, M.Kom selaku dosen pembimbing I dan pembimbing II, yang selalu sabar membimbing saya dalam pengerjaan skripsi ini, semoga ilmu yang beliau amalkan berguna bagi seluruh Umat Manusia.

Terima kasih kepada seluruh teman seperjuangan yang menemani dan banyak membantu selama kuliah. Teknik Informatika angkatan 2016, khususnya kepada Muhammad Arief Sulton, Damar Adyun, Nanda Nafisah dan Lutfia Miftahurrahma yang banyak memberi masukan dalam pengerjaan skripsi. Semoga kita semua mampu mewujudkan segala cita-cita yang kita impikan. Amin Allahumma Amin.

KATA PENGANTAR

Assalamu'alaikum Wr.Wb.

Segala puji bagi Allah SWT tuhan semesta alam, karena atas segala rahmat dan karunia-Nya sehingga penulis mampu menyelesaikan skripsi dengan baik dan lancar. Shalawat serta salam selalu tercurah kepada tauladan terbaik Nabi Muhammad SAW yang telah membimbing umatnya dari zaman kebodohan menuju islam yang *rahmatan lil alamiin*.

Dalam menyelesaikan skripsi ini, banyak pihak yang telah memberikan bantuan baik secara moril, nasihat dan semangat maupun materiil. Atas segala bantuan yang telah diberikan, penulis ingin menyampaikan doa dan ucapan terimakasih yang sedalam-dalamnya kepada:

1. Prof. Dr. Abdul Haris, M.Ag, selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Sri Harini, M.Si selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Cahyo Crys dian, selaku Ketua Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Irwan Budi Santoso, M.Kom., selaku dosen pembimbing I yang telah meluangkan waktu untuk membimbing, memotivasi, dan mengarahkan dan memberi masukan kepada penulis dalam pengerjaan skripsi ini hingga akhir.
5. Fathurrochman, M.Kom selaku dosen pembimbing II yang senantiasa memberi masukan dan nasihat serta petunjuk dalam penyusunan skripsi ini.

6. Segenap dosen Teknik Informatika yang telah memberikan bimbingan keilmuan selama masa studi.
7. Teman-teman seperjuangan Teknik Informatika angkatan 2016.

Berbagai kekurangan dan kesalahan mungkin pembaca temukan dalam penulisan skripsi ini, untuk itu penulis menerima segala kritik dan saran yang membangun dari pembaca sekalian. Semoga apa yang menjadi kekurangan bisa disempurnakan oleh peneliti selanjutnya dan semoga karya ini senantiasa dapat memberi manfaat. Amin.

Wassalamualaikum Wr.Wb

Malang, 17 Juni 2020

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
PERNYATAAN KEASLIAN TULISAN	iv
MOTTO	v
HALAMAN PERSEMBAHAN	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xii
ABSTRAK	xiii
ABSTRACT	xiv
ملخص	xv
 BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Pernyataan Masalah.....	6
1.3 Tujuan Penelitian.....	7
1.4 Manfaat Penelitian.....	7
1.5 Batasan Penelitian.....	7
 BAB II STUDI PUSTAKA	
2.1. Pengertian Lahan Kosong.....	8
2.2. <i>Convolutional Neural Network</i>	10
2.3. Prinsip <i>Convolutional Neural Network</i>	12
2.4. Desain Arsitektur CNN.....	13
2.5. <i>Pooling Layer</i>	16
2.6. <i>Activation Function</i>	17
2.6.1. Fungsi Aktivasi Linier.....	17
2.6.2. Fungsi Aktivasi Nonlinear.....	18
2.6.3. ReLU (<i>Rectrified Linear Unit</i>).....	19
2.7. Pengolahan Citra.....	20
2.7.1. Definisi Citra.....	20
2.7.2. Resolusi Citra.....	21
2.7.3. Resolusi Piksel.....	21
2.7.4. Ekstrasi Ciri Suatu Citra.....	22
2.8. Segmentasi.....	23
2.9. <i>Backpropagation</i>	24
2.10. Citra <i>Google Earth</i>	25
2.11. Penelitian Terkait.....	25
 BAB III METODOLOGI PENELITIAN	
3.1. Prosedur Penelitian.....	30
3.2. Pengumpulan Data.....	32
3.2.1. <i>Data Training Set</i> dan <i>Tasting Set</i>	36
3.3. Desain Sistem.....	36
3.3.1. Input Gambar.....	38

3.3.2.	<i>Preprocessing</i> Citra	38
3.3.3.	Implementasi CNN	41
3.3.3.1	Proses <i>Training</i>	41
3.3.3.2	Proses <i>Testing</i>	48
BAB IV HASIL DAN PEMBAHASAN		
4.1.	Langkah-langkah dan Uji Coba	50
4.2.	Hasil dan Uji Coba	50
4.2.2.	Hasil Proses <i>Training</i>	50
4.2.3	Hasil Proses <i>Testing</i>	52
4.2.4	Pengaruh Parameter Learning Terhadap Akurasi	54
4.2.4.1.	Pengaruh Jumlah <i>Epoch</i>	55
4.2.4.2.	Pengaruh Jumlah Nilai <i>Learning Rate</i>	55
4.2.4.3.	Pengaruh Ukuran Citra	56
4.3.	Pembahasan.....	57
BAB V PENUTUP		
5.1.	Kesimpulan	63
4.1.	Saran	63
DAFTAR PUSTAKA		
LAMPIRAN- LAMPIRAN		



DAFTAR GAMBAR

Gambar 2.1 <i>Architecture of CNN</i>	14
Gambar 2.2 Contoh kasus skripsi dengan CNN.....	14
Gambar 2.3 <i>Feature Map</i>	15
Gambar 2.4 <i>Linear Function</i>	17
Gambar 2.5 <i>Non-linear Activation Function</i>	18
Gambar 2.6 ReLU	19
Gambar 2.7 <i>Backpropagation</i>	24
Gambar 3.1 Prosedur Penelitian	30
Gambar 3.2 Kondisi objek Lahan kosong dari ketinggian 5000 ft	33
Gambar 3.3 Kondisi objek Lahan Hijau/RTH dari ketinggian 5000 ft	34
Gambar 3.4 Kondisi objek lahan sawah dari ketinggian 5000 ft	34
Gambar 3.5 Kondisi objek lahan Permukiman dari ketinggian 5000 ft	35
Gambar 3.6 Pengambilan <i>training set</i> dan <i>Tasting Set</i>	36
Gambar 3.7 Desain Sistem.....	37
Gambar 3.8 <i>Cropping Image</i>	39
Gambar 3.9 Augmentasi Gambar	40
Gambar 3.10 Skema Training dan Testing.	42
Gambar 3.11 Arsitektur CNN	43
Gambar 3.12 Simulasi Desain CNN model	44
Gambar 3.13 Konsep Cara Pengujian dataset	49
Gambar 3.14 Output Klasifikasi.....	49
Gambar 4.1 Grafik <i>accuracy</i> dan <i>loss</i>	52
Gambar 4.2 Pseudocode perhitungan akurasi	52
Gambar 4.3 Hasil <i>feature learning</i> tahap 1	58

DAFTAR TABEL

Tabel 2.1 Kriteria Lahan Kosong9
 Tabel 4.1 Hasil predeksi *training*51
 Tabel 4.2 Hasil prediksi proses testing54
 Tabel 4.3 Pengaruh *Epoch*55
 Tabel 4.4 Pengaruh nilai *learning rate*.....56
 Tabel 4.5 Pengaruh ukuran citra.....56



ABSTRAK

Alam, Islam Nur. 2020. **Identifikasi Lahan Kosong Kota Batu Berbasis Citra Google Earth Menggunakan Algoritma Convolutional Neural Network (CNN)**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Pembimbing : (I) Irwan Budi Santoso, M. Kom. (II) Fathurrochman, M.Kom

Kata Kunci : *neural nets, convolution, preprocessing, Google Earth image, training, testing, data augmentation.*

Identifikasi lahan kosong Kota Batu berbasis citra google earth adalah materi penelitian yang saat ini terus dikembangkan . Detailnya ciri yang mesti ditangkap membuat para peneliti berlomba-lomba menemukan metode yang paling cocok untuk melakukan identifikasi. Algoritma Convolutional Neural Network (CNN) menjadi salah satu algoritma yang saat ini paling diunggulkan dalam bidang klasifikasi dan identifikasi objek saat ini. Data yang digunakan pada penelitian ini ada data berupa citra hasil cropping dari citra google earth. Sebelum data dilatih, data akan dilakukan preprocessing antara lain crop, menggunakan citra RGB, kemudian diaugmentasi. Setelah itu algoritma akan dilatih menggunakan citra sebanyak 1300 citra yang dibagi menjadi 2 kelas. Dengan ukuran citra sebesar 32x32 pixel, kemudian dilatih dengan 1000 epoch dan learning rate 0.001. Setelah itu, model yang sudah dilatih dilakukan pengujian pada 300 citra baru. Dari penelitian tersebut didapatkan hasil berupa akurasi training sebesar 99,6% dan akurasi testing sebesar 86,34%.

ABSTRACT

Alam, Islam Nur. 2020. **Identification of Batu City Empty Land Based on Google Earth's Image Using the Convolutional Neural Network (CNN) Algorithm.** Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
Advisor : (I) Irwan Budi Santoso, M. Kom. (II) Fathurrochman, M.Kom

Keywords : *neural nets, convolution, preprocessing, Google Earth image, training, testing, data augmentation.*

Identification of vacant land based on Google Earth imagery is a research material that is currently being developed. The details of the characteristics that must be captured make researchers competing to find the most suitable method for identification. The Convolutional Neural Network (CNN) algorithm is one of the algorithms currently most favored in the field of classification and identification of objects at this time. The data used in this study there is data in the form of image cropping results from Google Earth imagery. Before the data is trained, the data will be preprocessed, including crop, using an RGB image, and then segmented. After that the algorithm will be trained using 1300 images divided into 2 classes. With an image size of 32x32 pixels, then trained with 1000 epochs and a learning rate of 0.001. After that, the model that has been trained is tested on 300 new images. From these studies the results obtained in the form of training accuracy of 99.6% and testing accuracy of 86.34%.

ملخص

عالم, اسلام نور. 2020. تحديد أرض باتو سيتي الفارغة بناءً على صورة Google Earth باستخدام خوارزمية الشبكة العصبية التلافيفية. البحث العلوم والتكنولوجيا بجامعة موالن مالك إبراهيم الإسلامية الحكومية مالنج.

مشرف : (I) اروان بودي سانتوسو, (II) فاطور الرحمن

كلمت البحث : زيادة البيانات, صورة رمادية, تجهيزها, شبكات العصبية, التفاف, اختبارات, تدريب

يعد تحديد الأراضي الشاغرة في مدينة باتو استنادًا إلى صور برنامج *Google Earth* مادة بحثية يتم تطويرها حاليًا. إن تفاصيل الخصائص التي يجب التقاطها تجعل الباحثين يتنافسون للعثور على أنسب طريقة لتحديد الهوية. تعد خوارزمية الشبكة العصبية التلافيفية واحدة من الخوارزميات الأكثر تفضيلًا حاليًا في مجال تصنيف وتحديد الأشياء في هذا الوقت. البيانات المستخدمة في هذه الدراسة توجد بيانات في شكل نتائج اقتصاص الصور من صور برنامج *Google Earth* قبل تدريب البيانات، سيتم معالجة البيانات مسبقًا، بما في ذلك الاقتصاص، باستخدام صورة RGB، ثم تقسيمها. بعد ذلك سيتم تدريب الخوارزمية باستخدام 1300 صورة مقسمة إلى فئتين. مع صورة بحجم 32x32 بكسل، ثم تدريبها على 1000 حقة ومعدل تعلم 0.001. بعد ذلك، تم اختبار النموذج الذي تم تدريبه على 300 صورة جديدة. من هذه الدراسات تم الحصول على نتائج في صورة دقة تدريب 99.6% ودقة اختبار 86.34%.

BAB I

PENDAHULUAN

1.1. Latar Belakang

Lahan kosong seringkali kurang mendapatkan perhatian di dalam penataan lahan perkotaan. Tanpa disadari, keberadaannya ternyata menimbulkan permasalahan serius dalam pengembangan lahan perkotaan. Pada dasarnya, lahan kosong mengindikasikan inefisiensi penggunaan lahan, terutama di perkotaan, dimana suplai lahan di perkotaan sangat terbatas, sementara itu perminataannya terus bertambah. Dengan adanya lahan kosong, berarti terdapat persil-persil lahan yang tidak dimanfaatkan secara optimal. Lebih lanjut, menurut Darin-Drabkin (1977:25) menyimpulkan bahwa “lahan kosong dapat mendorong peningkatan harga lahan secara cepat karena sediaan lahan menjadi berkurang, sedangkan permintaannya terus bertambah. Harga lahan yang terus meningkat tentunya akan menghambat kegiatan pembangunan yang akan dilakukan”.

Zaman ini eknologi pengolahan citra digital (*Digital Image Prosessing*) sangat berkembang pesat semenjak adanya teknologi komputer yang sanggup mengolah algoritma pada tahun 1960. Berbagai masalah-masalah yang bisa diselesaikan oleh salah satu fokus keilmuan di bidang informatika. Bukan hanya itu saja, teknologi pengolahan kosong citra digital saat juga mulai berkembang dan diterapkan pada berbagai bidang ilmu lainnya, mulai bidang hiburan dan kesehatan.

Dilihat dari perkembangannya, teknologi pengolahan lahan kosong citra ini semakin berkembang tidak hanya karena kebutuhan manusia akan teknologi yang lebih sederhana saja tapi juga kebutuhan sistem juga. Contoh kecil yang bisa

dilihat dalam sistem sebelumnya yaitu sistem absensi. Dahulu sistem absensi masih manual, mulai berkembang dengan memanfaatkan pengolahan citrayaitu dengan adanya *finger print*. Dari penemuan teknologi, *finger print* sangat membantu..karena cukup mudah digunakan meskipun tidak menggunakan operatorxuntuk mengisi absensi. Kemudian dari kebutuhan sistem, untuk mengenali sidik jari orang yang absen lalu mencocokkan dengan data yang sebelumnya sudah ada juga memerlukan metode yang bagus. Hal ini sangat penting karena jika terjadi kesalahan baik dalam mengenali ataupun mencocokkan data maka akan terjadi masalah dan kerugian oleh beberapa pihak. Oleh sebab itu, zaman ini tidak hanya teknologi dari citra saja yang berkembang tetapi metode-metode *recognition* yang melatar belakngan teknologi pengolahan lahan kosong citra juga berkembang.

Seperti yang sudah disampaikan sebelumnya, *metode recognition* sangat penting perannya ketika diaplikasikan sesuai dengan kebutuhannya. Karena akan banyak sekali masalah yang akan terselesaikan ketika metode *recognition* ini jika berhasil diaplikasikan jika memiliki peforma yang bagus. Hal ini juga harus didukung oleh perangkat komputasi yang bagus agar metoderecognition dapat bekerja dengan baik. Salah satu pengaplikasian metode *recognition* yang akan membantu manusia yaitu jika diaplikasikan pada bidang robotika.

Metode CNN merupakan “salah satu metode deep learning yang mampu melakukan proses pembelajaran mandiri untuk pengenalan objek, ekstraksi objek dan klasifikasi serta dapat diterapkan pada citra resolusi tinggi yang memiliki model distribusi nonparametric” (Zhang et al, 2018:5).

Tentunya ini bukanlah hal yang mudah, membutuhkan kemampuan *machine learning* yang sangat bagus untuk mewujudkannya. Karena jika kebutuhan sistem sampai pada taraf untuk mengenali persepsi, maka dibutuhkan metode pengenalan yang bagus, tapi juga bisa mengenali objek citra secara akurat. Hal ini tentunya akan menjadi tantangan tersendiri, karena untuk mengenali objek dari citra *google earth* hingga pada taraf identifikasi objek tentu membutuhkan sebuah algoritma yang mampu mengenali ciri suatu citra secara detail.

Bukan hanya itu saja, jika berpikir lebih luas lagi, tentu tahu bahwa citra dari peta *google earth* dari setiap objek berbeda. Oleh karena itu, dibutuhkan *dataset* besar yang berisi objek lahan kosong yang berbeda-beda berupa citra dari hasil *cropping google earth* pada setiap kategorinya. Supaya algoritma yang digunakan nanti memiliki variasi dan perbandingan data yang cukup banyak sehingga algoritma tidak kesulitan ketika menemukan objek-objek baru yang belum ada pada *databasenya*.

Demi menjangkau itu semua dibutuhkan sebuah *mechine learning* yang mampu mempelajari pekerjaan itu secara lebih dalam. Oleh karena itu, saat ini dikembangkan metode pembelajaran terbaru yang mampu mempelajari suatu pekerjaan secara lebih dalam yaitu *deep learning*. "*Deep learning* adalah bagian dari *machine lerning* yang memungkinkan komputer mempelajari berdasarkan pengalaman masa lalu dan memahami perintah berdasarkan konsep yang diberikan"(Goodfellow *et al.*, 2016:17). Hal yang membedakan *deep learning* dengan *machine lerning* adalah pada struktur pembelajarannya yang lebih dalam berdasarkan representasi data yang ingin dipelajari. "*Deep learning* saat ini menjadi topic hangat yang sering

dibicarakan terkait penelitian tentang *object recognition*, *object detection*, *speech recognition* dan berbagai macam *pattern recognition* lainnya”(Javier Ruiz *et al.*, 2018:6).

“*Deep learning* ini telah mengubah paradigma penelitian *pattern recognition* yang sebelumnya masih memisahkan antara *feature extraction* bersama dengan metode klasifikasi secara terpisah. Dengan adanya *deep learning* ini menggunakan keduanya dalam satu struktur bisa dilakukan bersama”(Albani *et al.*, 2017:9). Salah satu algoritma yang menerapkan metode *deep learning* adalah *deep neural network* atau lebih dikenal dengan *Convolutional Neural Network* (CNN).

Ide dasar algoritma *Convolutional Neural Network* ini adalah meniru struktur algoritma *neural network* yang melakukan proses pembelajaran melalui beberapa lapisan. Selain itu jika dilihat dari latar belakang adanya algoritma *neural network* ini adalah keinginan untuk meniru kemampuan berfikir manusia. Oleh karena itu kemampuan untuk mempelajari suatu data secara lebih dalam sangat cocok diterapkan pada algoritma *neural network* ini. Meskipun begitu, secara struktural algoritma CNN ini cukup berbeda dari algoritma *neural network*. Lebih tepatnya *neural network* digunakan pada sebagian kecil struktur algoritma CNN ini, yaitu pada bagian klasifikasinya. Sementara untuk proses *feature learning*, sesuai dengan namanya, algoritma ini menggunakan teori konvolusi untuk proses *feature learning*.

Dalam beberapa penelitian “*deep learning* telah menunjukkan performa yang luar biasa. Hal ini sebagian besar dipengaruhi oleh kemampuannya untuk mempelajari *dataset* yang besar dan teknik melatih jaringan yang lebih dalam.

Selain itu dengan dukungan perangkat komputasi yang lebih kuat membuat *deep learning* kini menjadi banyak digunakan”(Goodfellow et al., 2016:19). Kemampuan CNN di patok sebagai model terbaik untuk memecahkan permasalahan *object recognition*. Namun dalam CNN, seperti model *deep learnig* lainnya, memiliki kelemahan yaitu proses komputasi algoritma yang cukup lama. Tetapi dengan perkembangan hardware yang semakin pesat, hal tersebut dapat diselesaikan menggunakan teknologi *Graphical Procesing Unit* (GPU) dan PC yang memiliki spesifikas tinggi.

Berdasarkan latar belakang ini, penelitian ini akan mencoba untuk mengimplementasikan algoritma CNN ini. Pada penelitian ini berfokus terhadap seberapa besar akurasi algoritma CNN untuk mengidentifikasi lahan kosong Kota Batu dari citra hasil *cropping Google Earth*. Karena ini merupakan penelitian untuk mengidentifikasi lahan kosong, maka dibutuhkan objek yang akan digunakan sebagai bahan penelitian. Oleh karena itu, objek yang akan dijadikan bahan penelitian kali ini adalah lahan kosong Kota Batu dari citra hasil *cropping Google Earth*. Sebelum dilakukan *cropping* citra, diharuskan survei lokasi dari objek yang akan dijadikan dataset. Jadi nantinya citra tersebut nantinya akan diolah untuk dijadikan sebagai objek untuk penelitian ini.

Penelitian ini akan sangat bermanfaat sehingga dapat memudahkan pemerintah Kota Batu untuk mengenali lahan kosong kota batu dari citra google earth secara cepat, akurat dan *real time*. Hal ini selaras dengan sabda Rasulullah ﷺ bersabda:

خَيْرُ النَّاسِ أَنْفَعُهُمْ لِلنَّاسِ

“Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia” (HR. Ahmad, ath-Thabrani, ad-Daruqutni. Hadits ini dihasankan oleh al-Albani di dalam *Shahihul Jami* no:3289).

Menjadi seseorang yang bermanfaat untuk orang lain adalah salah satu karakter yang harus dimiliki seorang muslim. Karena pada dasarnya, ketika kita memberikan atau melakukan sesuatu yang bermanfaat untuk orang lain, maka manfaatnya akan kembali untuk kebaikan diri kita sendiri. Hal ini dijelaskan dalam firman Allah ﷻ pada surah al-isra’ ayat 7:

إِن أَنْسَحْتُمْ أَحْسَنْتُمْ لِأَنْفُسِكُمْ

“Kemudian kami katakan (Jika kalian berbuat baik) dengan mengerjakan ketaatan (berarti kalian berbuat baik bagi diri kalian sendiri)”(QS. Al-Isra:7).

1.2. Pernyataan Masalah

Berdasarkan latar belakang tersebut penelitian ini mengangkat masalah tentang seberapa besar akurasi metode *Convolutional Neural Network* (CNN) untuk mengidentifikasi lahan kosong Kota Batu berbasis citra *google earth*?

1.3. Tujuan Penelitian

Secara umum tujuan utama diadakannya penelitian ini adalah mengenali suatu objek. Adapun tujuan yang di dapat dari penelitian ini adalah untuk mengetahui seberapa besar akurasi algoritma *Convolutional Neural Network* (CNN) untuk dapat mengidentifikasi lahan kosong kota batu berbasis citra *google earth*.

1.4. Manfaat Penelitian

Manfaat yang diharapkan dari sistem identifikasi lahan kosong Kota Batu berbasis citra *google earth* menggunakan metode *Convolutional Neural Network* (CNN) ini adalah:

1. Penelitian ini bisa dikembangkan lagi pada bidang robotika untuk menentukan identifikasi lahan kosong berdasarkan ciri suatu citra dari *google earth*.
2. Penelitian ini bisa dikembangkan untuk mendeteksi lahan kosong secara real time pada *google earth* dan *moogle maps*.
3. Penelitian ini dapat memudahkan pemerintah Kota Batu untuk mengidentifikasi lahan kosong Kota Batu dari citra *google earth*.

1.5. Batasan Penelitian

Batasan yang ada pada penelitian ini adalah sebagai berikut:

- a. Data training yang digunakan hasil *cropping* dari foto satelit citra *google earth* khusus daerah Kota Batu.
- b. Sebelum *cropping* citra *google earth* diharuskan survei terlebih dahulu untuk semua lokasi lahan kosong Kota Batu.

BAB II

STUDI PUSTAKA

2.1. Pengertian Lahan kosong

Terdapat beberapa definisi yang mencoba menjelaskan pengertian tentang Lahan kosong, diantaranya adalah:

1. Kivell (1993: 151), mendefinisikan “lahan kosong sebagai lahan yang menurut pemerintah daerah setempat belum dimanfaatkan sesuai dengan fungsinya, yaitu fungsi yang mengacu pada rencana wilayah. Lahan kosong dapat berbentuk properti berupa tanah atau bangunan yang tidak dipergunakan”.

2. Chapin dan Kaiser (1979: 265) menyatakan bahwa “lahan kosong sebagai adalah sebidang lahan yang di atasnya secara fisik tidak terdapat bangunan, akan tetapi berpotensi untuk digunakan”.

3. Sensus Nasional Amerika, 1971, mendefinisikan lahan kosong sebagai lahan yang tidak dihuni pemiliknya, padahal secara fisik dapat dihuni. Pengertian ini juga mengacu pada bangunan-bangunan yang ditelantarkan oleh pemiliknya.

Sementara itu, definisi lahan kosong yang digunakan di Indonesia, dapat dilihat dari sumber-sumber berikut:

1. Buku Petunjuk Tata Cara Kerja Pengukuran Tanah (BPN 1992) menjelaskan lahan kosong sebagai lahan tidak terbangun yang sudah diperuntukkan atau diberi haknya tetapi tidak diusahakan sesuai dengan hak yang diberikan/ditelantarkan.

2. Permen Agraria/Kepala BPN No.3 Tahun 1998 tentang Pemanfaatan Tanah Kosong untuk Tanaman Pangan mendefinisikan lahan kosong sebagai lahan yang

tidak dimanfaatkan sesuai dengan sifat dan tujuan pemberian haknya atau Rencana Tata Ruang Wilayah yang berlaku.

3. PP No.36 Tahun 1998 tentang Penertiban dan Pendayagunaan Lahan Terlantar, menjelaskan definisi tentang lahan terlantar, yaitu lahan Hak Milik, Hak Guna Usaha, Hak Guna Bangunan, atau Hak Pakai yang dengan sengaja tidak dipergunakan oleh pemegang haknya sesuai dengan keadaannya atau sifat dan tujuan haknya atau tidak dipelihara dengan baik (Pasal 3).

Berdasarkan penjelasan-penjelasan Undang-undang dan para pakar, terlihat bahwa lahan kosong memiliki pengertian yang beragam. Namun demikian, pada dasarnya pengertian tersebut mengandung tiga variabel yang dapat menjadi karakteristik dari lahan kosong.

Tabel 2.1 Kriteria Lahan Kosong
(Sumber: Badan Pertahanan Nasional, 1992, 1998)

Kriteria	Pengertian
Lahan Tidak Terbangun	lahan kosong sebagai lahan tidak terbangun yang sudah diperuntukkan atau diberi haknya tetapi tidak diusahakan sesuai dengan hak yang diberikan/ditelantarkan(BPN, 1992).
Lahan Terlantar	menjelaskan definisi tentang lahan terlantar, yaitu lahan Hak Milik, Hak Guna Usaha, Hak Guna Bangunan, atau Hak Pakai yang dengan sengaja tidak dipergunakan oleh pemegang haknya sesuai dengan keadaannya atau sifat dan tujuan haknya atau tidak dipelihara dengan baik (Pasal 3)(BPN, 1998).
Lahan Tidur	Lahan yang sudah mendapatkan hak atau <u>ijin lokasi</u> dan sudah diperoleh <u>haknya</u> baik oleh perusahaan atau badan hukum, tetapi <u>belum digunakan sesuai dengan peruntukannya dalam ijin lokasi</u> (BPN, 1998).

Dari pengertian tersebut, lahan kosong dapat diidentifikasi berdasarkan kriteria berikut ini:

1. Lahan kosong terjadi pada lahan yang sudah dikuasai

Kriteria ini menjadi dasar untuk menentukan objek lahan kosong, dimana lahan kosong dapat terjadi pada lahan yang dikuasai dengan hak atas lahan (Hak Milik, Hak Guna Usaha, Hak Guna Bangunan, dan Hak Pakai), perijinan (Ijin Lokasi), atau penguasaan lainnya yang sah menurut peraturan.

2. Lahan kosong dapat berupa lahan terbangun dan tidak terbangun

Kriteria ini untuk menunjukkan lingkup lahan kosong, yaitu dapat berupa lahan terbangun maupun tidak terbangun.

3. Lahan yang tidak dimanfaatkan sesuai dengan sifat dan tujuan penguasaannya atau rencana tata ruang yang berlaku dalam waktu satu tahun sejak lahan tersebut dikuasai

Hal ini berarti jika selama satu tahun suatu lahan tidak dimanfaatkan, maka dapat dikatakan sebagai lahan kosong. Dasar perhitungannya dapat dilakukan sejak diperolehnya dasar penguasaan (sertifikat hak atas lahan atau ijin lokasi), jual-beli, atau sejak lahan tersebut ditelantarkan pemiliknya. Kriteria ini mengikat kedua kriteria sebelumnya sehingga suatu lahan dapat diidentifikasi sebagai lahan kosong.

2.2. Convolutional Neural Network

Pada tahun 1960 ketika Hubel dan Wiesel meneliti neuron yang digunakan untuk orientasi sensitif lokal-selektif dalam sistem visual kucing, mereka menemukan bahwa struktur jaringan khusus dapat secara efektif mengurangi kompleksitas jaringan saraf umpan balik dan kemudian mengusulkan

Convolutional Neural Network (CNN). *Convolutional Neural Network* adalah algoritma pengenalan yang efisien dan banyak digunakan dalam pengenalan pola dan pengolahan citra. “*Convolutional Neural Network* memiliki banyak fitur seperti struktur sederhana, parameter pelatihan yang kurang dan kemampuan beradaptasi. *Convolutional Neural Network* telah menjadi topik hangat dalam analisis suara dan pengenalan gambar. Bobotnya struktur jaringan bersama membuatnya lebih mirip dengan jaringan saraf biologis. Ini mengurangi kompleksitas model jaringan dan jumlah bobot” (Liu, 2015:16).

“Umumnya, struktur CNN mencakup dua lapisan satu adalah lapisan ekstraksi fitur, masukan setiap neuron terhubung ke bidang reseptif lokal dari lapisan sebelumnya, dan mengekstrak fitur lokal. Setelah fitur lokal diekstraksi, hubungan posisi antara itu dan fitur lainnya juga akan ditentukan. Yang lainnya adalah lapisan peta fitur, setiap lapisan komputasi jaringan terdiri dari sejumlah peta fitur. Setiap peta fitur adalah pesawat, berat neuron di pesawat sama. Struktur peta fitur menggunakan fungsi sigmoid sebagai fungsi aktivasi dari jaringan konvolusi, yang membuat peta fitur memiliki pergeseran invariance. Selain itu, karena neuron di pesawat pemetaan sama berbagi bobot, jumlah parameter bebas dari jaringan berkurang. Setiap lapisan konvolusi di jaringan syaraf konvolusi diikuti oleh lapisan komputasi yang digunakan untuk menghitung rata-rata lokal dan ekstrak kedua, dua lapisan ekstraksi fitur unik ini mengurangi resolusi”(Liu, 2015:18).

“CNN terutama digunakan untuk mengidentifikasi perpindahan, zoom dan bentuk inversi grafis dua dimensi yang distortif. Karena lapisan deteksi fitur CNN belajar dengan data pelatihan, ia menghindari ekstraksi fitur eksplisit dan secara

implisit belajar dari data pelatihan saat kita menggunakan CNN. Selanjutnya, neuron di bidang peta fitur yang sama memiliki bobot yang sama, sehingga jaringan bisa belajar secara bersamaan. Ini adalah keuntungan besar dari jaringan konvolusi sehubungan dengan jaringan neuron yang terhubung satu sama lain. Karena struktur khusus dari bobot berbagi lokal CNN membuatnya memiliki keunggulan unik dalam pengenalan suara dan pemrosesan gambar. Tata letaknya lebih dekat ke jaringan saraf sebenarnya. Bobot bersama mengurangi kompleksitas jaringan. Secara khusus gambar vektor input multi-dimensi dapat langsung masuk ke jaringan, yang menghindari kompleksitas rekonstruksi data dalam ekstraksi fitur dan proses klasifikasi”(Liu, 2015:20).

2.3. Prinsip *Convolutional Neural Network*

“Algoritma jaringan saraf konvolusi adalah perceptron multilayer yang merupakan desain khusus untuk identifikasi informasi gambar dua dimensi. Selalu memiliki lebih lapisan: lapisan masukan, lapisan konvolusi, lapisan sampel dan lapisan keluaran. Selain itu, dalam arsitektur jaringan yang dalam, lapisan konvolusi dan lapisan sampel dapat memiliki banyak. CNN tidak seperti mesin boltzmann yang dibatasi, harus sebelum dan sesudah lapisan neuron di lapisan yang berdekatan untuk semua koneksi, algoritma jaringan syaraf tiruan, setiap neuron tidak perlu melakukan citra global, hanya merasakan area lokal dari gambar. Selain itu, setiap

parameter neuron disetel sama, yaitu pembagian bobot, masing-masing neuron dengan kernel konvolusi yang sama dengan citra dekonvolusi”(Liu, 2015:20).

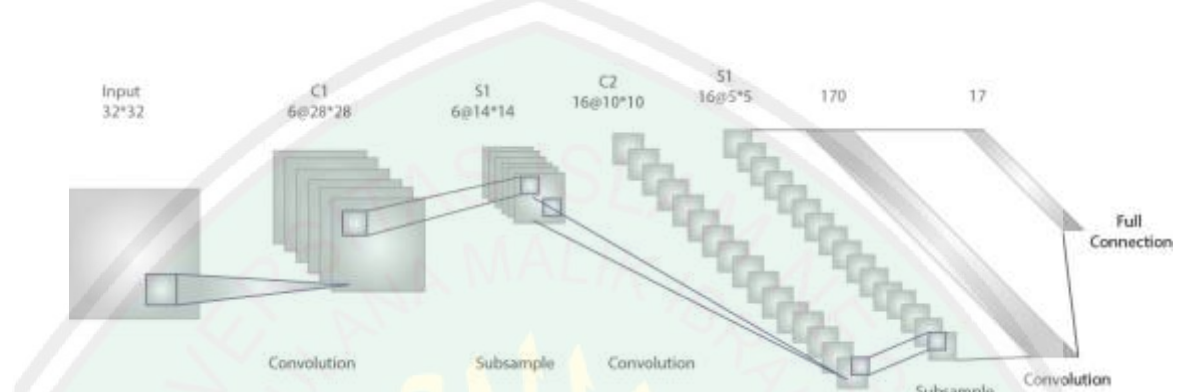
“Algoritma CNN memiliki dua proses: konvolusi dan sampling. Proses konvolusi: gunakan filter yang dapat dilatih F_x , dekonvolusi gambar masukan (Tahap pertama adalah gambar input, masukan dari konvolusi setelah adalah fitur gambar masing-masing layer, yaitu Feature Map), lalu tambahkan bias b_x , kita bisa mendapatkan lapisan konvolusi C_x . Proses sampling: n piksel setiap lingkungan melalui langkah penyatuan, menjadi piksel, dan kemudian dengan bobot skalar $W_x + 1$ tertimbang, tambahkan bias $b_x + 1$, dan kemudian oleh fungsi aktivasi, menghasilkan n kali lipat fitur peta $S_x + 1$ ”(Liu, 2015:25).

“Teknologi kunci CNN adalah field reseptif lokal, pembagian bobot, sub sampling berdasarkan waktu atau ruang, sehingga bisa mengekstrak fitur dan mengurangi ukuran parameter pelatihan. Keuntungan algoritma CNN adalah untuk menghindari ekstraksi fitur eksplisit, dan secara implisit untuk belajar dari data pelatihan. Bobot neuron yang sama pada permukaan pemetaan fitur, sehingga jaringan dapat dipelajari secara paralel, mengurangi kompleksitas jaringan. Mengadopsi struktur sub sampling berdasarkan waktu atau ruang, dapat mencapai tingkat ketahanan tertentu, skala dan deformasi perpindahan; Input informasi dan topologi jaringan bisa menjadi pertandingan yang sangat baik, Ini memiliki keunggulan unik dalam pengenalan suara dan pengolahan kosong gambar”(Liu, 2015:28).

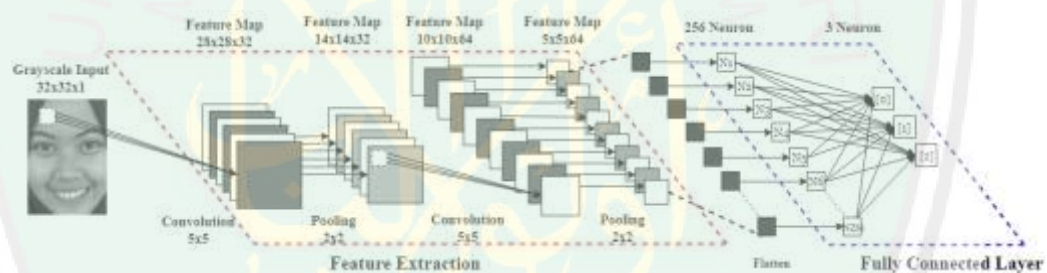
2.4. Desain Arsitektur CNN

“Algoritma CNN membutuhkan pengalaman dalam desain arsitektur, dan perlu melakukan debug tanpa henti dalam aplikasi praktis, agar bisa mendapatkan

arsitektur aplikasi CNN yang paling sesuai. Berdasarkan contoh pada gambar abu-abu dengan input 96×96 . Pada tahap preprocess, mengubahnya menjadi 32×32 dari ukuran gambar. Desain kedalaman lapisan 7 model konvolusi: lapisan input, lapisan konvolusi C1, lapisan sub sampling S1, lapisan konvolusi C2, lapisan sampling S2, lapisan tersembunyi H dan lapisan keluaran F”(Liu, 2015:77).



Gambar 2.1 Architecture of CNN



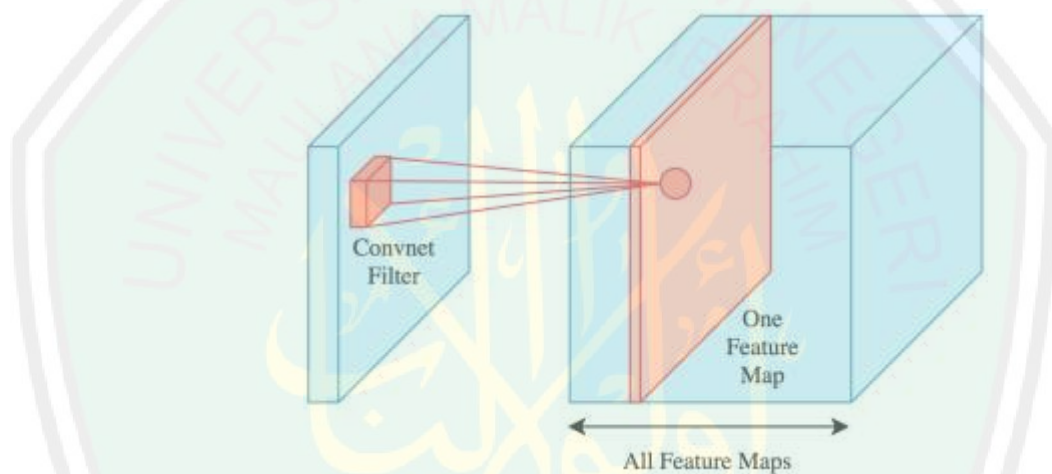
Gambar 2.2 Contoh kasus skripsi dengan CNN
(Sumber: Mubarok Hamdani, 2019)

Arsitektur dari CNN dibagi menjadi 2 bagian besar, yaitu:

a. *Feature Extraction Layer*

Proses yang terjadi pada arsitektur ini adalah melakukan *encoding* dari sebuah *image* menjadi features yang berupa angka-angka yang mempresentasikan image tersebut atau *feature extraction*. Pada Gambar 2.3 merupakan *channel* RGB (*Red, Green, Blue*) *image* berukuran 32×32 *pixels* yang sebenarnya adalah multidimensional *array* dengan ukuran $32 \times 32 \times 3$ (3 merupakan jumlah *channel*). *Convolutional layer* terdiri dari

neuron yang tersusun sehingga membentuk sebuah filter dengan panjang dan tinggi (*pixels*). Sebagai contoh, pada layer pertama pada *feature extraction layer* biasanya adalah *conv. layer* dengan ukuran $5 \times 5 \times 3$. Panjang 5 pixels , tinggi 5 pixels dan tebal/jumlah nya 3 buah sesuai dengan *channel* dari *image* tersebut. Ketiga filter ini akan digeser keseluruhan bagian dari gambar. Setiap pergeseran akan dilakukan operasi “dot” antara *input* dan nilai dari filter tersebut sehingga menghasilkan sebuah *output* atau biasa disebut dengan *activation map* atau *feature map*.



Gambar 2.3 *Feature Map* (Sumber: Dertat Arden, 2019)

b. *Fully Connected Layer*

Feature map yang dihasilkan dari feature extraction layer masih berbentuk multidimensional array, sehingga harus melakukan “flatten” atau reshape feature map menjadi sebuah vector agar bisa digunakan sebagai input dari fully-connected layer. Fully Connected Layer yang dimaksud disini adalah Multi Layer Perceptron yang sudah pernah dipelajari.

Lapisan Fully-Connected adalah lapisan di mana semua neuron aktivasi dari lapisan sebelumnya terhubung semua dengan neuron di lapisan selanjutnya seperti

halnya jaringan saraf tiruan biasa. Setiap aktivasi dari lapisan sebelumnya perlu diubah menjadi data satu dimensi sebelum dapat dihubungkan ke semua neuron di lapisan. Lapisan Fully-Connected biasanya digunakan pada metode Multi Layer Perceptron untuk mengolah data sehingga bisa diklasifikasikan. Perbedaan antara lapisan Fully-Connected dan lapisan konvolusi biasa adalah neuron di lapisan konvolusi terhubung hanya ke daerah tertentu pada input, sementara lapisan FullyConnected memiliki neuron yang secara keseluruhan terhubung. Namun, kedua lapisan tersebut masih mengoperasikan produk dot, sehingga fungsinya tidak begitu berbeda (Danukusumo, 2017:30).

2.5. Pooling Layer

“Pooling layer dilakukan untuk mengurangi varians, mengurangi kerumitan perhitungan dan ekstrak fitur tingkat rendah dari lingkungan sekitar. Pooling layer terdapat dua jenis yaitu Max pooling dan Average pooling. Max pooling mengekstrak fitur yang paling penting seperti tepi sedangkan average pooling lebih halus dibandingkan dengan max pooling”(Rahman, 2017:78).

“Meskipun keduanya digunakan untuk alasan yang sama, max pooling lebih baik untuk mengekstrak fitur ekstrem. Average pooling kadang-kadang tidak dapat mengekstrak fitur yang baik karena dibutuhkan semua untuk menghitung dan menghasilkan nilai rata-rata yang mungkin atau tidak penting untuk deteksi objek. Average pooling membawa semua ke dalam hitungan dan mengalirkannya ke lapisan berikutnya yang berarti semua nilai benar-benar digunakan untuk pemetaan fitur dan pembuatan keluaran yang merupakan perhitungan yang sangat umum. Jika tidak diperlukan semua masukan dari lapisan konvolusi maka akan mendapatkan akurasi yang buruk untuk average pooling. Pada klasifikasi objek

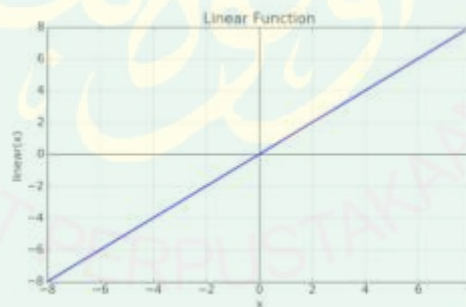
pooling layer yang banyak digunakan adalah average pooling. Pemilihan pooling layer tergantung pada jenis dataset yang digunakan”(Rahman, 2017:56)

2.6. Activation Function

“Activation function merupakan sebuah node yang ditambahkan ke akhir output dari setiap jaringan syaraf. Hal ini juga dikenal sebagai Transfer Function. Hal ini juga dapat dilampirkan di antara dua Neural Networks. Activation function digunakan untuk menentukan output jaringan syaraf tiruan. Activation function memetakan nilai yang dihasilkan di antara 0 sampai 1 atau -1 sampai 1 dan lainlain (tergantung pada fungsinya). Activation function pada dasarnya dapat dibagi menjadi 2 tipe yaitu fungsi aktivasi linier dan fungsi aktivasi non linier”(Sharma, 2017:80).

2.6.1. Fungsi Aktivasi Linier

Seperti yang dapat dilihat fungsinya adalah garis atau linier. Oleh karena itu, output dari fungsi tidak akan dibatasi antara suatu rentang.

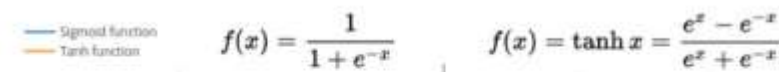


Gambar 2.4 *Linear Function* (Sumber: Samuel Sena, 2018)

“Fungsi aktivasi pada linear function memiliki persamaan : $f(x) = x$ dan Rentang : $-\infty$ sampai ∞ . Fungsi aktivasi linier ini tidak membantu dengan kompleksitas atau berbagai parameter data biasa yang diumpungkan ke jaringan syaraf tiruan”(Sharma, 2017:77).

2.6.2. Fungsi Aktivasi Nonlinear

Fungsi Aktivasi Nonlinear adalah fungsi aktivasi yang paling banyak digunakan. Grafik nonlinier dapat dilihat seperti gambar 2.5 ini:



The graph shows two non-linear activation functions. The Sigmoid function is represented by a blue curve that starts near 0 for negative x and approaches 1 for positive x. The Tanh function is represented by an orange curve that ranges from -1 to 1, passing through the origin. The equations for both functions are provided above the graph.

$$f(x) = \frac{1}{1 + e^{-x}}$$

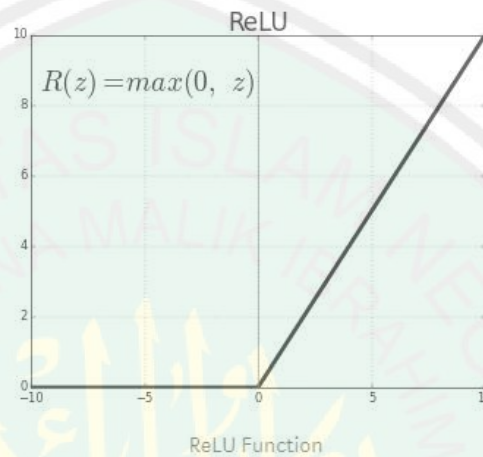
$$f(x) = \tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

Gambar 2.5 *Non-linear Activation Function* (Sumber: Samuel Sena, 2018)

“Fungsi aktivasi non linier memudahkan model untuk menggeneralisasi atau menyesuaikan dengan berbagai data dan untuk membedakan antara output. Terminologi utama yang perlu dipahami untuk fungsi nonlinier adalah Derivatif atau Diferensial dan Monotonik. Fungsi Derivatif atau Diferensial: Perubahan sumbu y w.r.t. Perubahan x -axis. Hal ini juga dikenal sebagai kemiringan dan fungsi Monotonik merupakan sebuah fungsi yang bervariasi sedemikian rupa sehingga tidak pernah berkurang atau tidak pernah meningkat. Terdapat beberapa fungsi aktivasi diantaranya adalah Sigmoid, Tanh, ReLU, dan Leaky ReLU”(Samuel Sena, 2018:24).

2.6.3. ReLU (*Rectrified Linear Unit*)

ReLU adalah fungsi aktivasi yang paling banyak digunakan di dunia saat ini. Karena, ini digunakan di hampir semua jaringan saraf konvolusi atau pembelajaran yang dalam.



Gambar 2.6 ReLU (Sumber: Samuel Sena, 2018)

Seperti yang dapat dilihat, ReLU setengah diperbaiki (dari bawah). Ini adalah $f(z)$ adalah nol bila z kurang dari nol dan $f(z)$ sama dengan z bila z tersebut atau sama dengan nol. Rentang: (0 sampai tak terbatas) Fungsi dan turunannya keduanya bersifat monoton. “Tapi masalahnya adalah semua nilai negatif menjadi nol segera yang menurunkan kemampuan model agar sesuai atau melatih dari data dengan benar. Itu berarti setiap masukan negatif yang diberikan pada fungsi aktivasi ReLU mengubah nilainya menjadi nol segera dalam grafik, yang pada gilirannya mempengaruhi grafik yang dihasilkan dengan tidak memetakan nilai negatif secara tepat” (Sharma, 2017:24).

2.7. Pengolahan Citra

Citra adalah suatu representasi (gambaran), kemiripan, atau imitasi dari suatu objek. Citra terbagi 2 yaitu citra yang bersifat analog dan ada citra yang bersifat digital. Citra analog adalah citra yang bersifat continue seperti gambar pada monitor televisi, foto sinar X, dan lain-lain. Sedangkan pada citra digital adalah citra yang dapat diolah oleh komputer (Sutojo, 2009:45).

2.7.1. Definisi Citra

“Citra atau image adalah suatu cahaya pada bidang dua dimensi. Ditinjau dari sudut pandang matematis, citra merupakan fungsi kontinu dari intensitas cahaya pada bidang dua dimensi. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan cahaya ini ditangkap oleh alat-alat optik, misalnya mata pada manusia, kamera, scanner, dan sebagainya, sehingga bayangan objek yang disebut citra tersebut terekam. Citra sebagai keluaran dari suatu sistem perekaman data dapat bersifat” (Iriyanto, 2014:33) :

- a. Optik berupa foto
- b. Analog berupa sinyal video seperti citra pada monitor televisi
- c. Digital yang dapat langsung disimpan pada suatu pita magnetic”

Citra dapat dikelompokkan menjadi dua bagian yaitu diam (*still image*) dan citra bergerak (*moving image*). Citra diam yang ditampilkan secara beruntun (sekuensial), sehingga memneri kesan pada mata sebagai gambar yang bergerak. Setiap citra didalam rangkaian tersebut *frame*. Gambar-gambar yang tampak pada film layar lebar atau pada hakekatnya terdiri dari ratusan sampai ribuan *frame*. (Sitorus, Syahrul dkk. 2006:24). Citra atau *image* juga dapat didefinisikan sebagai

fungsi dua dimensi $f(x,y)$ di mana x dan y adalah koordinat spasial dan amplitude f pada setiap pasang (x,y) disebut intensitas citra pada titik tersebut (Iriyanto, 2014).

2.7.2. Resolusi Citra

“Resolusi citra merupakan tingkat detail suatu citra. Semakin tinggi resolusi citra maka semakin tinggi pula tingkat detail dari citra tersebut. Satuan dalam pengukuran resolusi citra dapat berupa ukuran fisik berupa jumlah garis per mm/jumlah garis per inchi ataupun dapat juga berupa ukuran citra menyeluruh dengan jumlah garis tertinggi citra. Resolusi sebuah citra dapat diukur dengan berbagai cara sebagai berikut”(Putra, 2010:19) :

1. Resolusi Piksel
2. Resolusi spasial
3. Resolusi *spectral*
4. Resolusi temporal
5. Resolusi radiometric

2.7.3. Resolusi Piksel

Resolusi piksel merupakan perhitungan jumlah piksel dalam sebuah citra digital. Sebuah citra digital dengan tinggi N piksel dan lebar M piksel berarti memiliki resolusi sebesar $M \times N$. Resolusi piksel akan memberikan dua buah angka integer yang secara berurutan akan mewakili jumlah piksel lebar dan jumlah piksel tinggi dari citra tersebut.

“Pengertian lainnya dari resolusi piksel adalah merupakan hasil perkalian jumlah piksel lebar dan tingginya dan kemudian dibagi dengan 1 juta. Jenis resolusi piksel seperti ini sering kali dijumpai dalam kamera digital. Suatu citra

yang memiliki lebar 2.048 piksel dan tinggi 1.536 piksel maka akan memiliki total piksel sebanyak $2.048 \times 1.536 = 3.145.728$ atau 3,1 mega piksel. Perhitungan lainnya menyatakan dalam satuan piksel perinci. Satuan ini menyatakan banyaknya piksel yang ada sepanjang 1 inci baris dalam citra”(Putra, 2010:30).

2.7.4. Ekstraksi Ciri Suatu Citra

(Marques dan Furht 2002:13): Ekstraksi ciri merupakan proses pengindeksan suatu database citra beserta isinya. Secara matematik, setiap ekstraksi ciri merupakan encode dari vektor n dimensi yang disebut dengan vektor ciri. Komponen vektor ciri dihitung dengan pemrosesan citra dan teknik analisis serta digunakan untuk membandingkan citra yang satu dengan citra yang lain. Ekstraksi ciri diklasifikasikan menjadi 3 jenis, diantaranya low level, middle level, dan high level. Low level feature merupakan ekstraksi ciri berdasarkan isi visual seperti warna dan tekstur. Middle level feature merupakan ekstraksi berdasarkan wilayah citra yang ditentukan dengan segmentasi, sedangkan high level feature merupakan ekstraksi ciri berdasarkan informasi semantik yang terkandung dalam citra, macam-macam ekstraksi ciri:

a. Warna

Warna merupakan salah satu ciri visual yang digunakan dalam *Content Based Image Retrieval* (CBIR). Warna sangat baik jika digunakan untuk temu kembali citra karena memiliki hubungan yang sangat kuat dengan objek dalam sebuah citra, yang melatar belakangi gabungan background, skala, orientasi, perspektif dan ukuran.

b. Bentuk

Bentuk merupakan ciri dalam suatu citra yang sangat esensial untuk segmentasi citra karena dapat mendeteksi objek atau batas suatu wilayah. Proses yang dapat digunakan untuk menentukan ciri bentuk adalah deteksi tepi, threshold, segmentasi, dan perhitungan moment seperti (mean, median dan standard deviasi dari setiap lokal citra).

c. Tekstur

Tekstur merupakan ciri intrinsic dari suatu citra yang terkait tingkat kekasaran (roughness), granularitas (granulation), dan keteraturan (regularity) susunan piksel. Aspek tekstural dari sebuah citra dapat dimanfaatkan sebagai dasar dari segmentasi, klasifikasi, maupun interpretasi citra.

2.8. Segmentasi

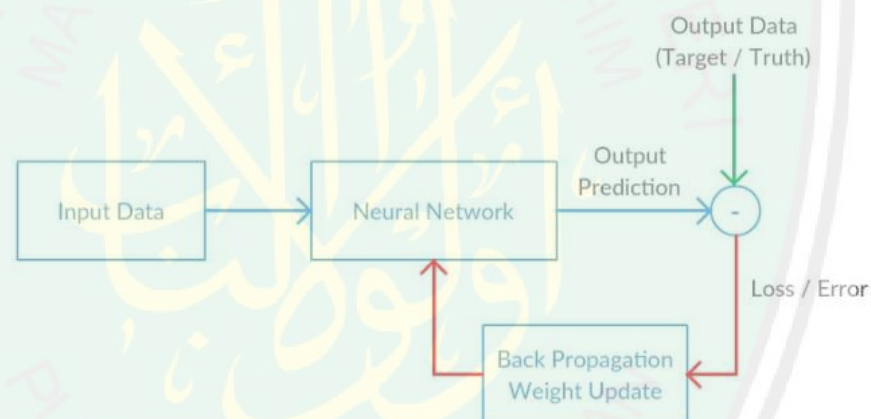
Menurut *Ramesh Jain, Kasturi, dan Schunk (1995:7)*, "segmentasi merupakan sebuah metode untuk membagi sebuah gambar menjadi sub-sub gambar yang disebut area (*region*). Ada 2 pendekatan yang dapat digunakan untuk membagi gambar-gambar menjadi daerah-daerah tertentu yaitu *Region-based* dan *Boundary estimation* menggunakan *edge detection*".

Pada pendekatan *region-based*, semua piksel yang berkorespondensi dengan sebuah objek dikelompokkan bersama dan diberikan *flag* yang menandakan bahwa mereka merupakan satu area. Dua prinsip yang penting dalam pendekatan ini adalah *value similiarity* dan *spatial proximity*. Dua piksel dapat dikelompokkan menjadi satu bila mempunyai karakteristik intensitas yang serupa atau bila keduanya memiliki jarak yang berdekatan. Sedangkan pada pendekatan *boundary estimation* menggunakan *edge detection*, segmentasi dilakukan dengan

menemukan piksel-piksel yang terletak pada sebuah batas are. Piksel tersebut (atau yang disebut sebagai edges) dapat diperoleh dengan melihat piksel-piksel yang berdekatan (*neighbouring pixels*).

2.9. Backpropagation

Salah satu metode yang digunakan dalam *neural network* dan yang paling sering digunakan dalam berbagai bidang aplikasi, seperti pengenalan pola, peramalan dan optimisasi adalah *backpropagation* karena metode ini menggunakan pembelajaran yang terbimbing. Algoritma ini menggunakan *error output* untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Dalam mendapatkan error ini, pertama harus melakukan *forward propagation*.



Gambar 2.7 *Backpropagation* (Sumber: Samuel Sena, 2018)

Arsitektur jaringan *backpropagation* seperti yang ditunjukkan pada Gambar 2.12 menunjukkan bahwa *neural network* terdiri dari tiga unit neuron pada *input layer* yaitu x_1 , x_2 , dan x_3 , dua *hidden layer* yaitu z_1 dan z_2 , serta 1 unit neuron pada *output layer*. Terdapat bobot yang menghubungkan neuron pada input layer dengan neuron pertama pada lapisan tersembunyi yaitu v_{11} , v_{21} , v_{31} . Bobot b_{11} dan b_{12} merupakan bobot bias yang menuju neuron pertama dan kedua pada lapisan tersembunyi. Bobot yang menghubungkan z_1 dan z_2 dengan neuron pada

lapisan output adalah w_1 dan w_2 . Bobot bias b_2 menghubungkan lapisan tersembunyi dengan lapisan *output*.

2.10. Citra Google Earth

Google Earth adalah program komputer yang menyajikan representasi 3D Bumi berdasarkan citra satelit. Program ini memetakan Bumi dengan melapiskan citra satelit, foto udara, dan data GIS ke bola dunia 3D, yang memungkinkan pengguna untuk melihat kota dan lanskap dari berbagai sudut. Pengguna dapat menjelajahi dunia dengan memasukkan alamat dan koordinat, atau dengan menggunakan *keyboard* atau *mouse*. Program ini juga dapat diunduh pada smartphone atau tablet, menggunakan layar sentuh atau stylus untuk bernavigasi. Pengguna dapat menggunakan program untuk menambahkan data mereka sendiri menggunakan Keyhole Markup Language dan mengunggahnya melalui berbagai sumber, seperti forum atau blog. *Google Earth* dapat menampilkan berbagai jenis gambar yang dihamparkan di permukaan bumi dan juga klien Layanan Peta Web

2.11. Penelitian Terkait

Penelitian mengenai “Deep Learning untuk Deteksi Tanda Nomor Kendaraan Bermotor Menggunakan Algoritma *Convolutional Neural Network* dengan Python dan Tensorflow” yang dilakukan oleh Imam Taufiq (2018:23). Dalam penelitian ini menggunakan algoritma *Convolutional Neural Network* untuk klasifikasi dan mendeteksi plat kendaraan bermotor pada sebuah gambar. Pada penelitian ini terdapat 502 dataset gambar dan menggunakan perbandingan 80% untuk training serta 20% untuk testing. Proses training membutuhkan lebih dari 25.000 step dengan jumlah batch 8 dan pada saat batch yang digunakan adalah 4 membutuhkan 100.000 step sampai model yang di training mampu

mendeteksi keberadaan TNKB serta menghasilkan akurasi sekitar 99% pada sebuah gambar plat kendaraan bermotor.

Penelitian mengenai “object recognition with deep learning applied to fashion items detection in images” yang dilakukan oleh Helder Filipe de Sousa Russa (2017). Dalam penelitian ini menggunakan metode Fast R-CNN untuk mengklasifikasikan dan mendeteksi item fashion tertentu yang digunakan oleh orang pada sebuah gambar. Pada penelitian dilakukan 3677 train image per kategori dan melakukan 696 testing image per kategorinya. Hasil menunjukkan bahwa dengan menggunakan metode CNN untuk mendeteksi fashion item yang dipakai oleh seseorang menghasilkan rata-rata precision of close sebesar 78%, untuk pants sebesar 65% dan untuk rata-rata aksesoris seperti glasses sebesar 57%. Metode Fast R-CNN digunakan untuk lebih mempersingkat waktu dalam pelatihan objek.

Penelitian mengenai “klasifikasi citra menggunakan *convolutional neural network* pada Caltech 101” yang dilakukan oleh I Wayan Suartika E.P., dkk (2016). Penelitian ini menggunakan “metode deep learning untuk mengklasifikasikan unggas diantaranya dengan kategori emu, flamingo, ibis, pigeon, dan roaster yang terdiri dari 150 citra. Selain kategori tersebut dihasilkan pula 3 kategori yaitu cougar, crocodile, dan face. Hasil dari 5 kategori unggas menunjukkan bahwa persentase keberhasilan 20% sedangkan untuk 3 kategori lainnya menunjukkan persentase keberhasilan 50%. Dengan menggunakan data training yang baik dan optimal, maka subset dari data training tersebut juga akan menghasilkan klasifikasi yang baik.”

Penelitian mengenai “Object Detection for Autonomous Driving Using Deep Learning” yang dilakukan oleh Victor Vaquero Gomez (2015). Penelitian ini menggunakan *Convolutional Neural Network*. Penelitian ini memberikan hasil pengujian pada deteksi objek seseorang pada sebuah video yang menghasilkan akurasi yang cukup tinggi yaitu sebesar 89% dari Equal Error Rate.

Penelitian mengenai “Implementasi Metode *Convolutional Neural Network* untuk Klasifikasi Tanaman pada Citra Resolusi Tinggi” yang dilakukan oleh Erlyna Nour Arrofiqoh dan Harintaka. Dalam penelitian ini menggunakan algoritma *Convolutional Neural Network* untuk klasifikasi tanaman pada citra resolusi tinggi. Pada penelitian ini, diterapkan algoritma CNN untuk membedakan jenis tanaman dengan memberikan label semantik dari objek jenis tanaman. Penelitian menggunakan 5 kelas jenis tanaman, yaitu kelas tanaman padi, bawang merah, kelapa, pisang dan cabai. Proses learning jaringan menghasilkan akurasi 100% terhadap data training. Pengujian terhadap data validasi menghasilkan akurasi 93% dan akurasi terhadap data tes 82%. Hasil penelitian ini menunjukkan bahwa penggunaan metode CNN berpotensi untuk pendekatan pengenalan objek secara otomatis dalam membedakan jenis tanaman sebagai bahan pertimbangan interpreter dalam menentukan objek pada citra.

Penelitian mengenai “Deteksi Non-RTH(Ruang Terbuka Hijau) Kota Malang Berbasis Citra Google earth dengan Menggunakan Naïve Bayes Classifier” yang dilakukan oleh Irwan Budi Santoso. Studi kasus dalam penelitian ini adalah Kota Malang berdasarkan image atau citra satelit adalah Naïve Bayes Classifier (NBC). Hasil uji coba dengan menggunakan sampel pengujian,

menunjukkan tingkat akurasi metode tersebut dalam mendeteksi objek non-RTH kota Malang adalah 81%.

Penelitian mengenai “Deep Learning untuk Deteksi Tanda Nomor Kendaraan Bermotor Menggunakan Algoritma *Convolutional Neural Network* dengan Python dan Tensorflow” yang dilakukan oleh Imam Taufiq (2018). Dalam penelitian ini menggunakan algoritma *Convolutional Neural Network* untuk klasifikasi dan mendeteksi plat kendaraan bermotor pada sebuah gambar. Pada penelitian ini terdapat 502 dataset gambar dan menggunakan perbandingan 80% untuk training serta 20% untuk testing. Proses training membutuhkan lebih dari 25.000 step dengan jumlah batch 8 dan pada saat batch yang digunakan adalah 4 membutuhkan 100.000 step sampai model yang di training mampu mendeteksi keberadaan TNKB serta menghasilkan akurasi sekitar 99% pada sebuah gambar plat kendaraan bermotor.

Penelitian mengenai “Object Detection for Autonomous Driving Using Deep Learning” yang dilakukan oleh Victor Vaquero Gomez (2015). Penelitian ini menggunakan *Convolutional Neural Network*. Penelitian ini memberikan hasil pengujian pada deteksi objek seseorang pada sebuah video yang menghasilkan akurasi yang cukup tinggi yaitu sebesar 89% dari Equal Error Rate.

Berdasarkan penelitian yang dipublikasikan sebelumnya, diketahui bahwa belum ada penelitian mengenai pendeteksian objek khususnya pada objek gambar melalui citra google earth dengan mengidentifikasi lahan kosong menggunakan metode *Convolutional Neural Networks (CNN)*. Oleh karena itu, pada penelitian ini akan dilakukan pendeteksian dan pengklasifikasian objek lahan kosong menggunakan *Convolutional Neural Network*.

Dalam penelitian ini menggunakan algoritma Convolutional Neural Networks (CNN) yang digunakan merupakan metode CNN yang dikembangkan oleh Andre Lopes et al., (2017). Metode *Convolutional Neural Networks* (CNN) tersebut merupakan metode pengembangan dari model yang sudah pernah dibuat sebelumnya oleh Liu Kuang et al., (2016). Perbedaannya adalah Andre Lopes et al., (2017) ini berusaha untuk membentuk model CNN yang bagus dengan jumlah data yang lebih sedikit. Dengan menggunakan jumlah 2 layer konvolusi untuk proses *feature learning* dan 2 *full connect layer* untuk proses *image classification*, Dengan ukuran *input image* 32x32.



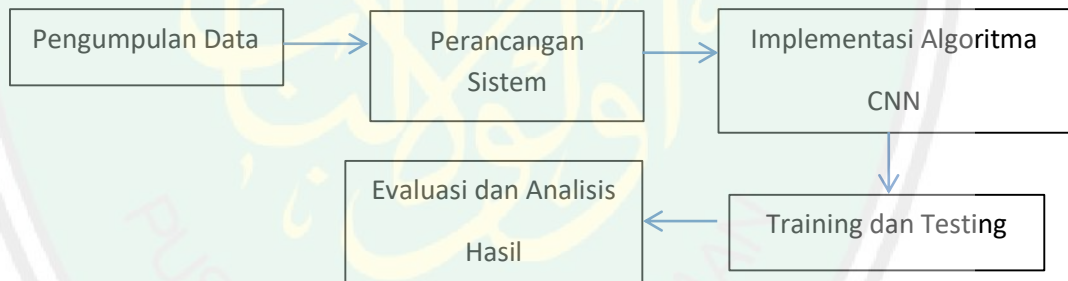
BAB III

METODE PENELITIAN

Pada bab ini akan dijelaskan tentang analisa dan perancangan sistem dari penelitian ini. Terdapat beberapa tahapan, yaitu tahapan penelitian yang dilakukan, kebutuhan sistem yang akan dibuat dan penyelesaian masalah identifikasi objek lahan kosong kota Batu berbasis citra *google earth* menggunakan algoritma *Convolutional Neural Network* (CNN).

3.1. Prosedur Penelitian

Dalam melakukan penelitian ini, untuk mempermudahnya maka dijabarkan langkah-langkah apa saja yang akan diambil dalam melakukan penelitian ini. Prosedur dari penelitian ini direpresentasikan pada gambar 3.1 ini.



Gambar 3.1 Prosedur Penelitian

Berdasarkan pada gambar 3.1 ini dapat dimengerti bahwa langkah-langkah apa saja yang harus dilakukan untuk mendapatkan hasil dari penelitian. Langkah pertama yang dilakukan adalah pengumpulan data untuk penelitian, data ini berupa data *training* dan *testing*. Bab tinjauan pustaka disajikan bukan hanya sebagai informasi atau materi tambahan saja, tetapi itu juga sebagai dasar penelitian ini dengan penelitian lainnya. Sehingga hal

tersebut bisa sebagai pembelajaran untuk membantu berjalannya penelitian ini. Langkah selanjutnya yang harus dikerjakan adalah proses perancangan sistem, pada proses sistem yang akan dibangun pada penelitian ini akan didefinisikan jalannya sistem yang akan dibangun pada penelitian ini. Setelah bagian ini telah selesai didefinisikan, selanjutnya bagaimana mengimplementasikan metode dan algoritma yang akan digunakan pada sistem ini. Seperti judul penelitian ini akan menggunakan salah satu algoritma *deep learning* yaitu *Convolutional Neural Network* (CNN). Pada penelitian ini algoritma *Convolutional Neural Network* (CNN) akan diimplementasikan untuk mengenali objek benda yang diinputkan pada sistem. Algoritma CNN ini dipilih karena merupakan salah satu algoritma *deep learning* yang memiliki tingkat akurasi yang cukup tinggi. Oleh karena itu penulis berharap dengan menggunakan metode ini, dapat membangun sistem yang memiliki kecerdasan untuk mengenali benda dengan akurasi tinggi layaknya kemampuan manusia.

Setelah metode berhasil diimplementasikan pada sistem yang dibangun ini, maka akan dilakukan *training* dan *testing* pada sistem. *Training* pada sistem ini bertujuan untuk memperkenalkan dan melatih sistem yang telah dibangun tersebut untuk mengenali benda yang akan diujikan nanti. Dengan melakukan *training* dengan sejumlah data *training* yang telah disiapkan, sistem kemudian akan siap untuk dilakukan uji coba untuk mengenali benda. Pada tahap *testing*, penulis berharap sistem akan mampu untuk mengenali benda yang dimaksudkan dan menampilkan akurasi ketepatan pengenalan benda. Akurasi yang didapatkan akan dijadikan sebagai acuan utama untuk selanjutnya, yaitu tahap pengembangan parameter. Pada tahap ini

penulis akan mencoba untuk mengembangkan beberapa parameter yang mungkin bisa mempengaruhi akurasi sistem. Hal ini dilakukan bertujuan untuk membandingkan beberapa parameter dan mendapatkan akurasi yang paling tinggi. Evaluasi dan analisis dari hasil uji coba ini diperlukan untuk membahas dan mengetahui seberapa besar kemampuan sistem dalam melakukan identifikasi lahan kosong melalui citra *google earth*.

3.2. Pengumpulan Data

Pada penelitian kali ini, untuk keperluan indentifikasi lahan kosong maka diperlukan objek berupa hasil *cropping* data *image* (citra) yang diambil dari *Google Earth* khusus untuk kawasan Kota Batu tahun 2019 dengan ketinggian sekitar 5000 ft. Data yang diperoleh dari *Google Earth* tersebut, akan digunakan sebagai data *training* untuk membangun struktur model pengenalan dengan menggunakan metode *Convolutional Neural Network* (CNN) beserta parameter modelnya, dan digunakan sebagai data uji coba (*testing*) untuk menguji metode dan aplikasi yang telah dibangun. Data *image* objek nyata yang diambil melalui *cropping* citra kawasan Kota Batu lahan kosong . Sedangkan untuk lahan kosong yang akan diidentifikasi pada penelitian ini dibagi menjadi 4 kategori, yaitu:

1. Ruang Terbuka Hijau
2. Lahan kosong Sawah
3. Perumahan
4. Lahan kosong Kosong

Untuk memvalidasi bahwa lahan kosong yang ditampilkan sudah benar dan sesuai maka langkah selanjutnya menentukan beberapa ciri yang sesuai dengan

kategori lahan kosong tersebut. Berikut penjelasan tekstual ciri dari 4 kategori objek yang digunakan pada penelitian kali ini:

- a. Lahan kosong Kosong, Chapin dan Kaiser (1979: 265) menyatakan bahwa lahan kosong sebagai adalah *sebidang lahan kosong yang di atasnya secara fisik tidak terdapat bangunan, akan tetapi berpotensi untuk digunakan.*

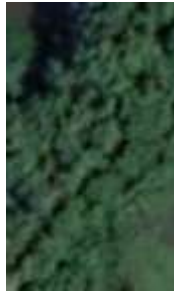
Dari penjelasan tersebut kemudian melakukan observasi dan memotret daerah objek tersebut dan melakukan *cropping* dari citra *Google Earth* dengan ketinggian sekitar 5000 ft dan berikut hasilnya.



Gambar 3.2 Lahan kosong kosong dari ketinggian 5000 ft
(Sumber: *Google Earth*, 2020)

- b. Ruang Terbuka Hijau(RTH), sebagai infrastruktur hijau perkotaan adalah bagian dari ruang-ruang terbuka (*open spaces*) suatu wilayah perkotaan yang diisi oleh tumbuhan, tanaman, dan vegetasi (endemik, introduksi) guna mendukung manfaat langsung dan/atau tidak langsung yang dihasilkan oleh RTH dalam kota tersebut yaitu keamanan, kenyamanan, kesejahteraan, dan keindahan wilayah perkotaan tersebut.

Dari penjelasan tersebut kemudian melakukan observasi dan memotret daerah objek tersebut dan melakukan *cropping* dari citra *Google Earth* dengan ketinggian sekitar 5000 ft dan berikut hasilnya.



Gambar 3.3 Ruang Terbuka Hijau(RTH)dari ketinggian 5000 ft (Sumber: *Google Earth*, 2020)

- c. Lahan kosong Sawah,”Tanah sawah adalah tanah yang digunakan untuk bertanam padi sawah, baik terus-menerus sepanjang tahun maupun bergiliran dengan tanaman palawija. Istilah tanah sawah bukan merupakan istilah taksonomi akan tetapi istilah ini lebih merupakan salah satu jenis penggunaan tanah yang dikelola sedemikian rupa untuk budidaya tanaman padi dan juga merupakan istilah umum seperti halnya, tanah hutan, tanah perkebunan, dan sebagainya”(Prasetyo, et al., 2004:57).

Dari penjelasan tersebut kemudian melakukan observasi dan memotret daerah objek tersebut dan melakukan *cropping* dari citra *Google Earth* dengan ketinggian sekitar 5000 ft dan berikut hasilnya.

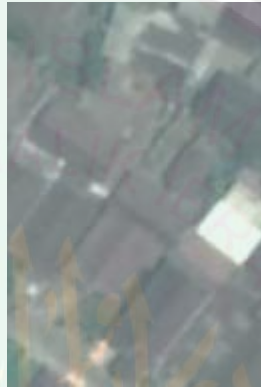


Gambar 3.4 lahan kosong sawah dari ketinggian 5000 ft (Sumber: *Google Earth*, 2020)

- d. Lahan kosong Permukiman, Permukiman merupakan bentuk tatanan kehidupan yang di dalamnya mengandung unsur fisik dalam arti permukiman merupakan

wadah aktifitas tempat bertemunya komunitas untuk berinteraksi sosial dengan masyarakat

Dari penjelasan tersebut kemudian melakukan observasi dan memotret daerah objek tersebut dan melakukan *cropping* dari citra Google Earth dengan ketinggian sekitar 5000 ft dan berikut hasilnya.



Gambar 3.5 lahan Permukiman dari ketinggian 5000 ft
(Sumber: *Google Earth*, 2020)

Dataset image objek nyata yang diambil melalui *cropping* citra google earth kawasan Kota Batu adalah lahan kosong , lahan hijau/RTH, sawah, dan lahan permukiman. Citra google earth cukup *uptodate* dengan keadaan lahan real yang menjadi objek penelitian. Oleh karena itu harus disertai dengan data-data perbandingan antara data lahan sebenarnya dengan *google earth* berikut ini tabel terdapat pada **Lampiran 1.**(Tabel Perbandingan Data Lahan Sebenarnya dengan hasil *Cropping* Citra *Google Earth*). Data berupa gambar hasil *cropping* citra Google Earth tersebut didapatkan 1300 gambar kemudian dibagi menjadi 2 bagian yaitu sebagai data *training* dan data *testing* dengan porsi kurang lebih 80%:20%.

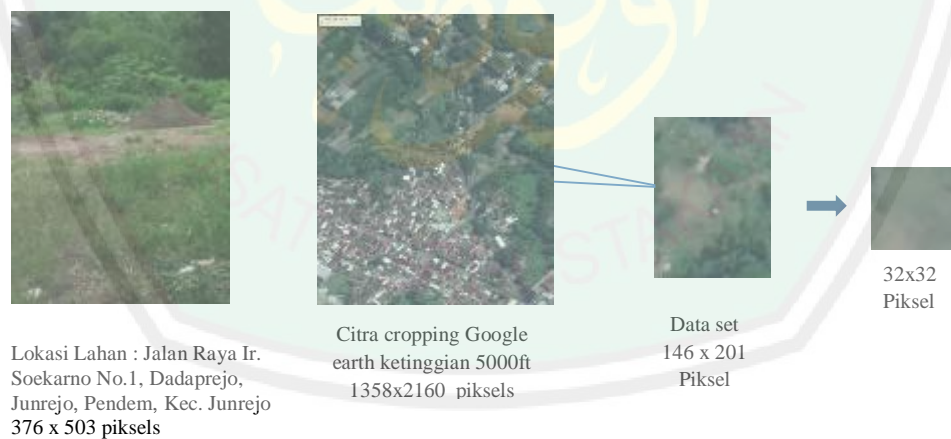
kemudian memisahkan data sesuai dengan kelasnya. Ketika dijalankan maka sistem akan otomatis membaca semua gambar yang ada pada *path* yang sudah ditentukan, sekaligus membaca juga merubah setiap folder yang ada pada *path* tersebut menjadi kelas yang disusun dalam bentuk *array class*.

3.2.1. Data Training Set dan Tasting Set

Data *training set* dan *Tasting Set* merupakan data di ambil dari cropping citra google earth. Proses pengumpulan *training set* dan *Tasting Set* harus memprioritaskan tentang kevalidasian data tersebut. Proses kevalidasian pengambilan *training set* dan *Tasting Set* dengan melakukan langkah-langkah sebagai berikut:

1. melakukan survei lokasi lahan di kota Batu
2. melakukan cropping dari citra *Google Earth*

Proses dapat dilihat dari gambar berikut:

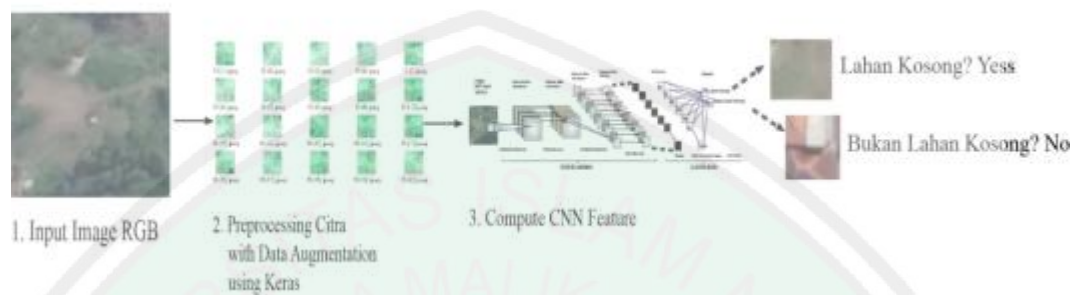


Gambar 3.6 Pengambilan *training set* dan *Tasting Set*
(Sumber: *Google Earth*, 2020)

3.3. Desain Sistem

Sebelum sistem penelitian ini dibangun, perlu adanya sebuah desain dari sistem yang akan dibangun terlebih dahulu. Desain sistem ini akan menjadi

gambaran besar seperti apa jalannya sistem yang akan dibangun nantinya. Dari desain sistem itu akan terlihat pada bagian mana algoritma yang CNN nanti akan diimplementasikan. Secara sederhana desain dari sistem identifikasi lahan kosong berbasis citra *Google Earth* ini tergambar pada gambar 3.8 ini.



Gambar 3.7 Desain Sistem

Secara sederhana pada penelitian ini, sistem yang akan dibangun nantinya terdiri dari tiga bagian utama, yaitu :

1. *Input* Gambar
2. *Preprocessing* Citra
3. Implementasi CNN
4. *Output*

Seperti yang dijelaskan pada subab sebelumnya yaitu pengumpulan data, data yang sudah dikumpulkan akan menjadi *input* yang akan digunakan. *Dataset* tersebut nantinya akan diproses pada bagian *image processing* dengan algoritma CNN. Baru kemudian, setelah pemrosesan selesai, akan menghasilkan *output* berupa hasil identifikasi lahan kosong dari hasil *inputan* citra berbasis *Google Earth*. Dari bagian utama sistem ini, pada masing-masing bagian terdapat sendiri-sendiri. Berikut akan dijabarkan pemrosesan terjadipada setiap bagian.

3.3.1. *Input Gambar*

Hasil dari *cropping* Gambar dari citra *Google Earth* yang sebelumnya sudah dikumpulkan, kemudian akan digunakan sebagai masukkan pada system yang akan dibangun. *Inputatn* gambar pada sistem ini dibagi menjadi dua *inputan*, yang pertama *inputan* gambar sebagai data *training* dan *inputan* gambar sebagai data *testing*.

3.3.2. *Preprocessing Citra*

Merujuk pada penelitian yang dilakukan oleh Irwan Budi Santoso., (2016) tentang Deteksi Non-RTH(Ruang Terbuka Hijau) Kota Malang Berbasis Citra Google Earth dengan *Naïve Bayes Classifier*(NBC) menggunakan 100 sampel *image* objek non-RTH yang terdiri dari 20 sampel *image* objek RTH rumah tinggal, 20 *image* objek gedung/kantor, 20 *image* objek lapangan, 20 *image* objek sawah dan 20 *image* objek jalan dengan akurasi **81 %**, maka pada penelitian kali ini menggunakan objek lahan kosong untuk diidentifikasi Berbasis Citra Google Earth dengan menggunakan metode *Convolutional Neural Network* (CNN). Dalam penelitian ini yang dilakukan, sebelum citra dijadikan *inputtan* untuk *training*, citra diolah terlebih dahulu agar lebih memudahkan algoritma CNN untuk melakukan pelatihan dan menemukan ciri dari citra yang dimasukkan.

Ada 3 tahap *preprocessing* citra yang dilakukan sebelum citra tersebut diolah oleh algoritma CNN :

a. *Cropping*

Cropping image objek dari citra sehingga terkumpul sampel *training* untuk setiap objek kelas. *Cropping* dilakukan untuk mendapatkan lebih banyak objek lahan kosong dari citra *Google Earth* dibandingkan objek selain lahan

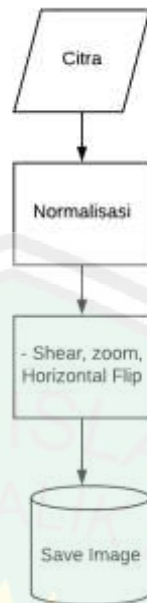
kosong. Proses *cropping* menjadi hal yang cukup penting, karena jika tidak dilakukan *cropping*, maka semua objek yang ada pada citra akan dianalisis oleh computer, padahal objek yang ingin diteliti adalah berupa lahan kosong dari citra *Google Earth*. Sehingga melakukan *cropping* pada bagian *background* dan objek selain lahan kosong. Jadi, hanya mengambil citra pada bagian lahan kosong saja. Untuk proses *cropping* ini masih dilakukan secara manual, karena harus setiap objek dari titik lokasi lahan kosong tertentu dari citra *Google Earth* untuk dilakukan *cropping*.



Gambar 3.8 *Cropping Image (Google Earth, 2020)*

b. Augmentasi Citra

kemudian dilakukan proses data augmentasi. Seperti yang diketahui, untuk mendapatkan performa optimal, *deep learning* membutuhkan data dengan jumlah yang banyak. Karena variasi objek yang ada di lokasi penelitian dan kemampuan pengambilan data cukup terbatas, maka dilakukan data augmentasi untuk memperbanyak variasi data. Data augmentasi adalah sebuah teknik memanipulasi sebuah data tanpa kehilangan inti atau esensi data-data tersebut. Augmentasi yang dilakukan *rescale*, *rotate*, *zoom* dan *flip*.



Gambar 3.9 Augmentasi Gambar

“Augmentasi citra pada python bisa dilakukan dengan menggunakan *library* ImageDataGenerator yang sudah disediakan oleh keras. Setelah melakukan import package, fungsi baru bisa ImageDataGenerator dipanggil, kemudian menentukan augmentasi apa saja yang akan diaplikasikan ke gambar. Untuk memanggil fungsi yang digunakan untuk melakukan augmentasi citra. Augmentasi citra yang dilakukan pada gambar adalah berupa”(Hamdani, 2019:49):

- a. *Rescalae* Citra: fungsi dari $\text{rescale}=1. / 255$ adalah untuk mernolmalisasi citra yaitu dengan membagi piksel terkecil dengan piksel terbesar. Fungsi ini akan dikenakan pada semua gambar sebelum melakukan augmentasi dengan fungsi lainnya.
- b. *Shear*: fungsi dari $\text{shear_range}=0.2$ adalah untuk menggeser citra searah jarum jam dengan pergerseran sebanyak 0.2 derajat.

- c. *Zoom*: fungsi dari `zoom_range=0.2` adalah untuk memperbesar citra dengan perbesaran sebanyak $1+0.2$ dari luas gambar.
- d. *Rotation*: fungsi dari `rotation=30` adalah untuk memutar gambar dengan sudut 30 derajat secara acak.
- e. *Flip*: fungsi dari `horizontal_flip=True` adalah untuk membalik gambar secara horizontal.

Semua fungsi augmentasi citra tersebut akan dikombinasikan dan diaplikasikan secara acak kepada semua citra.

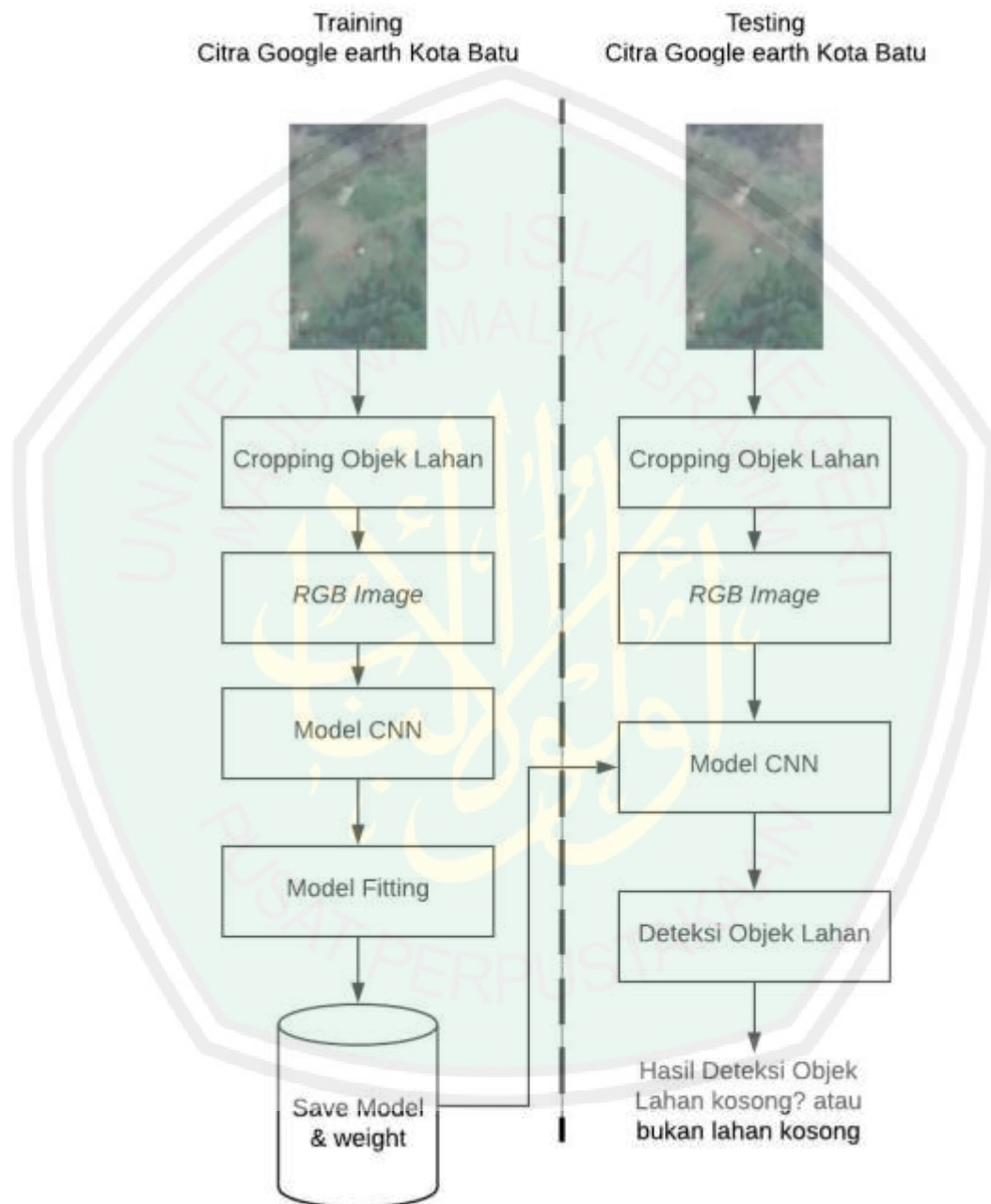
3.3.3. Implementasi CNN

Sebelum algoritma CNN diimplementasikan untuk mengidentifikasi citra lahan kosong, algoritma CNN perlu dilatih terlebih dahulu untuk menemukan ciri yang bisa dikenali pada gambar tersebut dan mengaktifkan *neuron* untuk klasifikasi. Pada tahap *training* pun, pengolahan kosong gambar dibagi menjadi 2 model yaitu, model *training* dan model *testing*. Setelah model dilatih, arsitektur CNN tersebut nantinya disimpan untuk kemudian dipanggil pada tahap *testing*.

3.3.3.1. Proses *Training*

Dalam sistem identifikasi lahan kosong Kota Batu berbasis citra *Google Earth* ini, untuk mendapatkan akurasi pengenalan objek yang tinggi maka algoritma perlu dilatih terlebih dahulu dengan sejumlah data *training*. Tujuan dari melatih algoritma ini adalah untuk menemukan ciri dari setiap gambar kemudian menandai *neuron-neuron* mana yang akan diaktifkan ketika gambar diklasifikasi. Oleh karena itu, perlu dibuat skema atau model untuk melakukan pelatihan pada algoritma agar ketika dilakukan..pengujian *object recognition*

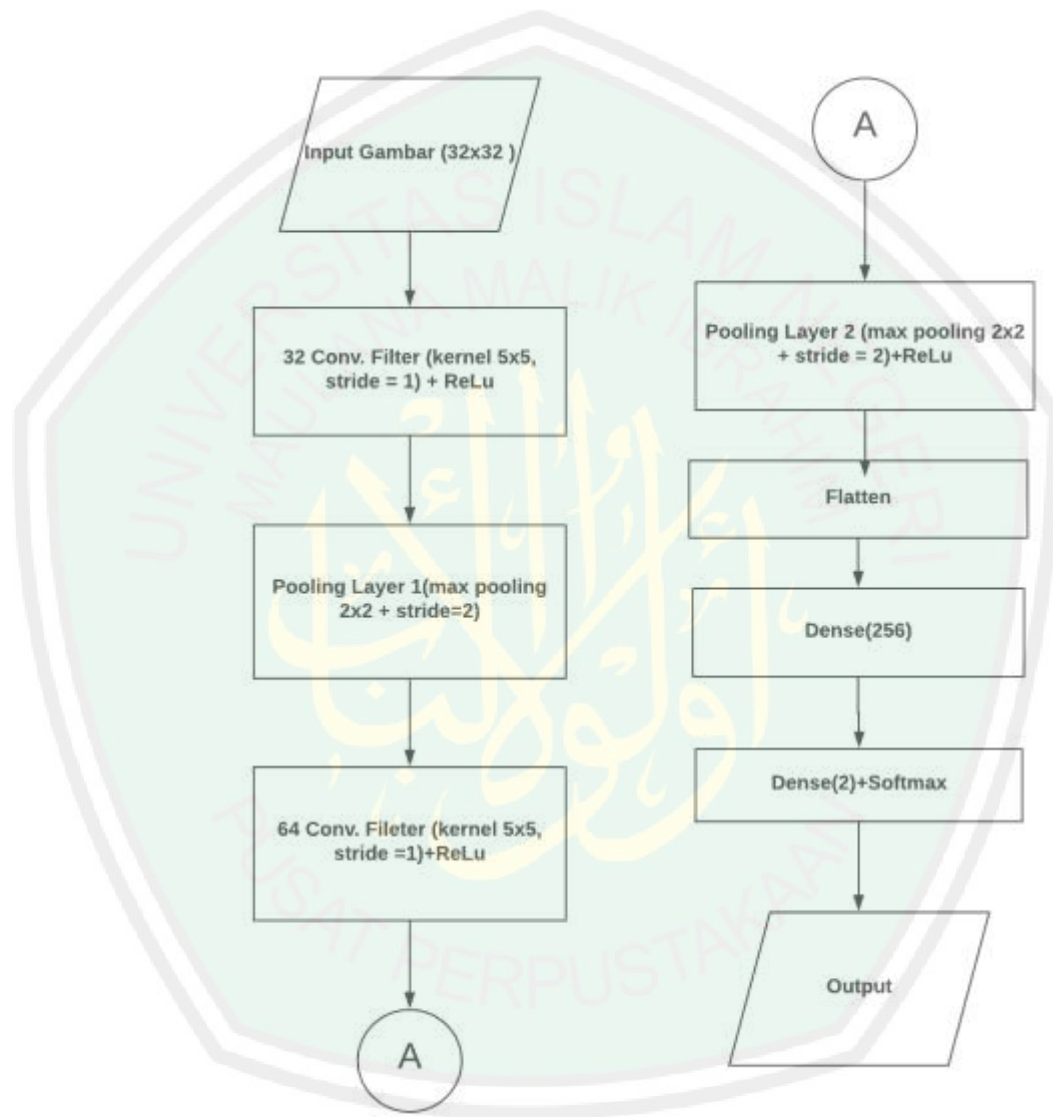
, algoritma sudah..terlatih. Berikut skema..pembentukan model yang akan dilakukan untuk..mendapatkan pola menggunakan algoritma CNN.



Gambar 3.10 Skema Training dan Testing

Sebelum melakukan pemrosesan gambar menggunakan algoritma CNN, data *training* yang dibutuhkan harus dipanggil terlebih dahulu. Kemudian,

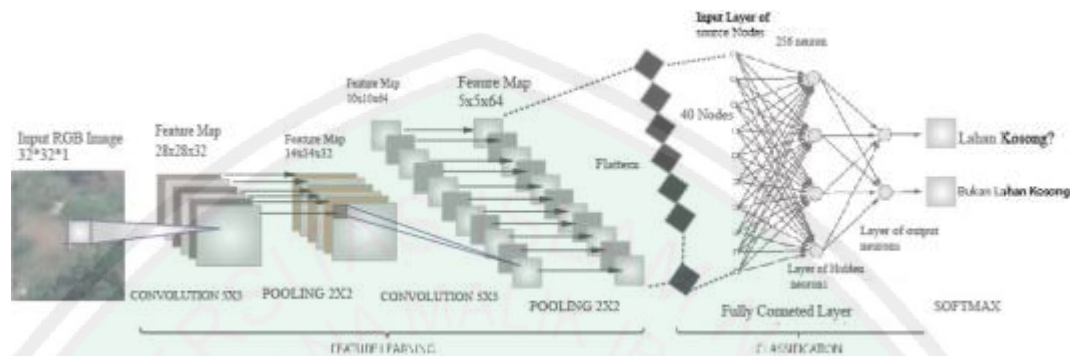
algoritma CNN diimplementasikan untuk melakukan *feature learning* dan klasifikasi gambar. Tapi sebelum itu, arsitektur dari algoritma CNN ini perlu dirancang terlebih dahulu. Berikut rancangan arsitektur CNN pada penelitian sebagai berikut:



Gambar 3.11 Blok Diagram Arsitektur CNN

Blok diagram arsitektur 3.11 merupakan model algoritma CNN yang dikembangkan oleh Andre Lopes *et al.*, (2017). Arsitektur tersebut merupakan arsitektur pengembangan dari arsitektur yang sudah pernah dibuat sebelumnya

oleh Liu Kuang *et al.*, (2016) tersebut. Perbedaannya adalah Andre Lopes *et al.*, (2017) ini berusaha untuk membentuk model CNN yang bagus dengan jumlah data yang lebih sedikit. Berdasarkan model pada blok diagram 3.13, arsitektur CNN dapat disimulasikan seperti pada gambar berikut:



Gambar 3.12 Desain Arsitektur CNN

Agar lebih mudah untuk dipahami arsitektur pada blok diagram 3.13, bisa dijelaskan sebagai berikut:

1. Sebelum gambar diinputkan, gambar akan diaugmentasi seperti yang sudah dijelaskan pada bagian *input* gambar 3.11. Hasil augmentasi tersebut disimpan dalam bentuk variable.
2. Variabel yang berisi hasil augmentasi citra, kemudian dipanggil untuk dijadikan *input* pada model CNN ini. *Input* citra yang akan digunakan berukuran 32x32x1.
3. Setelah itu citra yang sudah diinputkan tersebut masuk pada proses konvolusi pertama. Pada konvolusi pertama, gambar akan dikalikan dengan *kernel* berukuran 5x5 dengan *filter* sebanyak 32 *filter*. Proses perkalian dilakukan dengan menggeser *kernel* sebanyak 1 *stride*. Dari proses konvolusi pertama ini *feature map* yang dihasilkan adalah 28x28x32. Setelah itu *feature map* yang dihasilkan akan dilakukan aktivasi fungsi menggunakan ReLu.

4. Proses *pooling*, *pooling* merupakan proses subsampling pada *feature map* hasil konvolusi. Pada dasarnya *pooling* ini adalah melakukan pengurangan ukuran *feature map* dengan menggunakan *kernel* ukuran tertentu yang akan secara digeser pada seluruh area *feature map*. Pada penelitian ini menggunakan *max-pooling* dengan ukuran *kernel* 2x2 dan *stride* sebesar 2 piksel.
5. Setelah dilakukan *pooling*, *feature map* hasil *pooling* akan dijadikan sebagai *inputan* kembali untuk proses konvolusi kedua. Pada proses konvolusi kedua ukuran *kernel* yang digunakan masih sama yaitu 5x5, dengan *filter* yang digunakan bertambah menjadi 64 *filter*. Penambahan jumlah *filter* ini dilakukan karena pada proses *pooling* informasi yang dibuang semakin banyak, oleh karena itu penambahan *filter* dilakukan agar variasi informasi yang diperoleh dari inputan informasi yang tersedia semakin bertambah. Pada proses konvolusi kedua ini akan dihasilkan *featurex map* berukuran 10x10x64 sama seperti sebelumnya, proses konvolusi kedua ini juga menggunakan fungsi aktivasi ReLu.
6. Proses selanjutnya masuk ke proses *pooling* yang kedua, proses ini hampir sama dengan proses *pooling* yang pertama, yaitu dengan menggunakan *kernel* berukuran 2x2 dengan *stride* 2 piksel. Dari proses *pooling* yang kedua ini dihasilkan *feature map* berukuran 5x5x64.
7. Sebelum masuk pada proses *fully connected layer* (FC Layer). *Feature map* yang berupa 3D array akan dirubah menjadi *vector* 1-D list terlebih dahulu, proses ini sering dikenal *flatten*. Dari *feature map* berukuran 5x5x64 maka akan didapatkan nilai *vector* sebesar 1,250

piksel, Hasil ini akan dijadikan sebagai *inputan* pada proses *fully*”*connect ed layer*.

8. Setelah tahap *flatten*, maka akan diteruskan ke jaringan *fully connected layer* (FC Layer). Pada jaringan FC Layer ini menggunakan jaringan MLP dengan 4 *layer* yang terdiri dari *input layer*, *output layer* dan 2 *hidden layer*. Pada *input layer* terdiri dari 1 *neuron* hasil *flatten* dan *output layer* terdiri 2 *neuron*. Karena *dataset* terdiri dari 2 kelas, maka 2 *neuron* ini merupakan jumlah klasifikasi citra untuk setiap kelasnya. Sedangkan pada 2 jaringan *hidden layer* memiliki jumlah *neuron* yang berbeda, *hidden layer* pertama terdiri dari 120 *neuron*. Kemudian pada *hidden layer* kedua terdiri dari 84 *neuron*.

Proses pembentukan arsitektur CNN bisa dilakukan dengan mengimport *library* yang sudah disediakan yaitu *keras*. Kemudian mendefinisikan *layer-layer* yang diperlukan untuk membentuk arsitektur CNN.

fungsi *activation*=’*relu*’ atau Fungsi ReLU adalah $f(x) = \max(0, x)$. Ini diterapkan untuk output dari beberapa fungsi lain, seperti produk vektor-matriks. Salah satu cara ReLU meningkatkan jaringan saraf adalah dengan mempercepat pelatihan. Komputasi gradient (0 atau 1 tergantung pada tanda x). setiap elemen negatif diatur ke 0,0 - tanpa eksponensial, neuron 1 dengan output softmax menghasilkan ReLU pada output neuron 1, yang merupakan input dari neuron 2 kemudian neuron 2 dengan output softmax sehingga input neuron 2 pada dasarnya adalah ReLU (softmax (x1)).x

fungsi *activation*=’*softmax*’ menghitung probabilitas dari setiap kelas target atas semua kelas target yang memungkinkan dan akan membantu untuk

menentukan kelas target untuk input yang diberikan. Keuntungan utama menggunakan Softmax adalah rentang probabilitas output dengan nilai 0 hingga 1, dan jumlah semua probabilitas akan sama dengan satu. Softmax menggunakan eksponensial (e-power) dari nilai input yang diberikan dan jumlah nilai eksponensial dari semua nilai dalam input.

Fungsi `model.add(Dropout(0.5)) dropout` (dengan $p = 0,5$) digunakan pada masing-masing lapisan *layer* yang terhubung sepenuhnya sebelum output; itu tidak digunakan pada lapisan konvolusional. Ini menjadi konfigurasi yang paling umum digunakan. Regularisasi jaringan syaraf dimana beberapa neuron dipilih secara acak dan tidak dipakai selama pelatihan, neuron yang dibuang akan diberhentikan sementara jaringan dan bobot baru juga tidak diterapkan pada neuron pada saat melakukan *backpropagation*.

Agar model berjalan sesuai dengan parameter learning yang ditentukan, maka model CNN yang sudah disusun tersebut perlu untuk dikonfirmasi menggunakan `method model.compile()`.

“`Method model.compile()`. Merupakan untuk menyusun algoritma yang sudah dibuat agar melakukan *training* sesuai dengan parameter learning yang diberikan. Parameter *learning* yang diinisialisasi pada kode 3.16 adalah parameter *learning rate* dengan optimizer *Stochastic Gradient Descent* (SGD). SGD adalah metode optimasi *mechine learning* yang berguna untuk melakukan *update* bobot pada setiap parameter yang di *training*. Karena kebanyakan parameter yang harus di *training* pada jaringan FC *Layer*, maka optimizer SGD dirasa cocok untuk digunakan. Selain itu merujuk pada penelitian yang dilakukan oleh Aisha Wil” (Aisha C. Wilson, 2018:56). Ketika melakukan komparasi antara beberapa

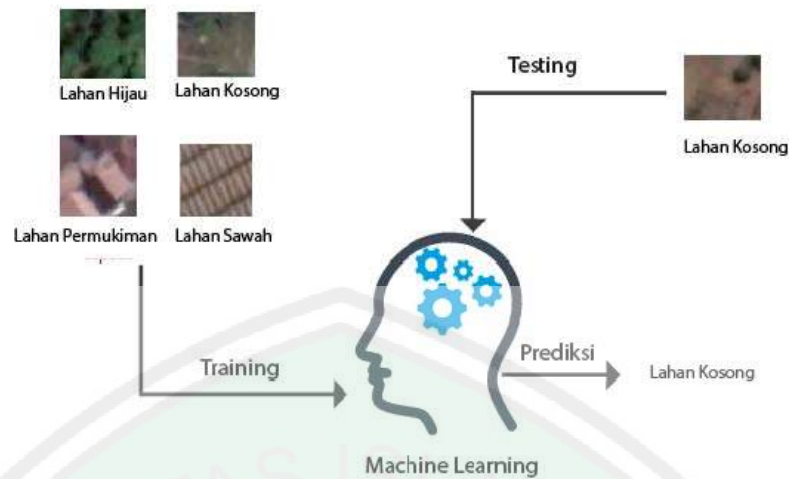
metode optimisasi *machine learning*, menunjukkan bahwa SGD memiliki performa paling bagus diantara optimasi lainnya.

Dari arsitektur CNN pada gambar 3.12 maka *dataset* akan dilakukan proses *training* sesuai dengan *training* model yang sudah dibuat sebelumnya. Proses untuk melakukan pelatihan model ini sering disebut dengan *model fitting*. Proses *model fitting* ini bisa dilakukan dengan secara memanggil method `model.fit()`, kemudian memasukkan variable data dan parameter iterasi yang harus dijalankan selama proses *model fitting* tersebut.

Ketika *model fitting* ini dijalankan, maka arsitektur CNN yang sudah dibentuk akan langsung bekerja untuk menguji data yang sudah dipersiapkan sebelumnya. Proses pelatihan dilakukan sejumlah *epoch* yang sudah ditentukan. Setiap menjalankan iterasi, sistem akan secara otomatis menampilkan *accuracy value* dan *loss value* dari data *training*. Setelah proses *training* selesai, model yang terbentuk dari proses *training* ini kemudian disimpan yang kemudian dipanggil saat proses *testing*.

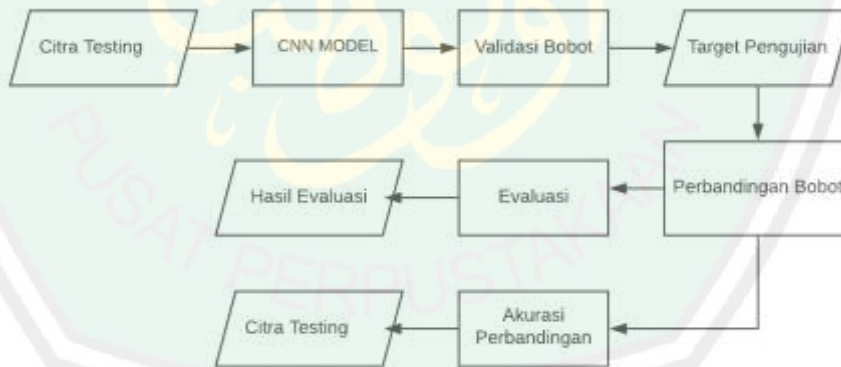
3.3.3.2. Proses *Testing*

Setelah model CNN melalui proses *training*, model tersebut akan di *testing* untuk menguji seberapa baik kinerja dari model yang sudah *training* sebelumnya. Proses pengujian ini dilakukan dengan cara mengujikan image baru yang belum pernah *training* sebelumnya oleh model tersebut. Dengan begitu, akan terlihat seberapa baik kinerja model yang sudah dibuat untuk mengidentifikasi. Step pertama yang akan adalah dengan memanggil model yang sudah dibentuk sebelumnya pada saat proses *training*.



Gambar 3.13 Konsep Cara Pengujian dataset

Proses selanjutnya adalah memunculkan hasil evaluasi dari *image* yang diujikan. Untuk menghasilkan hasil evaluasi, sistem kemudian membandingkan bobot antara *image* yang akan dievaluasi dengan bobot *output* pada model yang sudah dipanggil tersebut. Setelah bobot *image* tervalidasi, maka *image* tersebut akan diklasifikasikan ke *output* dengan bobot terdekat.



Gambar 3.14 Output Klasifikasi

Hasil dari pengujian ini nantinya berupa hasil klasifikasi dan nilai *probability* dari 2 bobot *output* yang terdekat dengan *image* yang diuji.

BAB IV

HASIL DAN PEMBAHASAN

4.1. Langkah-langkah Uji Coba

Langkah-langkah uji coba untuk mendeteksi lahan kosong berbasis citra *Google Earth* menggunakan algoritma CNN dapat dilihat sebagai berikut:

- a. Proses *Training*, dimana pada tahap ini akan dilakukan pelatihan pada data yang sudah disiapkan sebelumnya menggunakan model CNN. Model yang sudah terlatih nantinya, kemudian digunakan untuk mengukur seberapa bagus algoritma dalam melakukan identifikasi.
- b. Proses testing, tahap ini merupakan tahap yang cukup penting karena pada tahap ini program yang sudah dibangun dan *training* sebelumnya dilakukan pengujian. Data yang sudah disiapkan untuk diuji kemudian diujikan menggunakan model yang sudah dilatih sebelumnya.
- c. Pengembangan parameter *learning*, pada tahap ini akan ditampilkan perbandingan akurasi yang didapatkan dengan merubah beberapa parameter *learning* seperti *epoch*, *learning rate* dan ukuran citra.

4.2. Hasil dan Uji Coba

Berikut hasil uji coba pada sistem identifikasi lahan kosong berbasis citra *Google Earth* menggunakan algoritma CNN.

4.2.2. Hasil proses *Training*

Salah satu bagian terpenting dari berhasilnya proses identifikasi lahan kosong berbasis citra *Google Earth* ini adalah bagusnya hasil dari proses *training* ini. Baiknya hasil dari proses *training* akan memberikan dampak yang sangat tinggi terhadap hasil yang didapatkan pada proses uji coba nantinya.

Setelah arsitektur..dibentuk dan dilakukan proses model *fitting*, maka algoritma akan berlangsung melakukan pelatihan pada data yang//sudah disiapkan sebelumnya. Seperti yang sudah..dibahas pada sub bab sebelumnya, data yang akan digunakan untuk proses *training* adalah.sebesar 84.30% data. Data tersebut akan dieksekusi oleh algoritma CNN untuk diekstraksi yang dipelajari fiturnya.

Parameter iterasi yang dilakukan pada proses *training* ini sebanyak 1000 *epoch* dengan nilai *bact size* sebesar 20. Jadi proses pelatihan akan berlangsung dan diulang-ulang 1000 kali untuk memperoleh ekstrasi ciri dari suatu fitur yang dibutuhkan. Kemudian untuk nilai *learning rate* yang digunakan untuk melakukan *update* bobot setiap kali algoritma melakukan proses *backward-pass*.

Setelah proses *training* ini selesai, kemudian dilakukan *confusion matrix* untuk menghitung berbagai *performance metrics* untuk mengukur kinerja model yang telah dibuat.

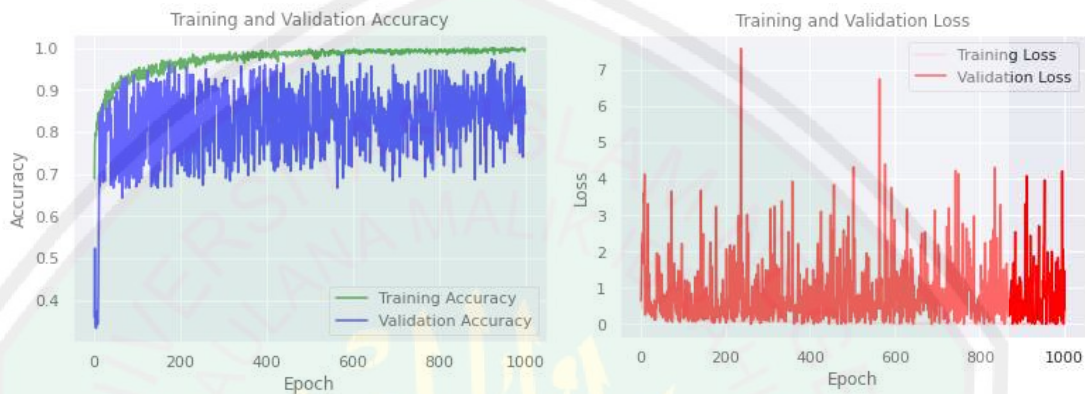
Seperti yang sudah dijelaskan proses *training*, dengan *confusion matrix* kita dapat mengetahui keakuartan dari model yang dibuat dengan *performance metrics* seperti: accuracy, recall, dan precision. Kemudian perhitungan akurasi untuk mendapatkan nilai akurasi, dilakukan dengan memasukkan hasil prediksi seperti pada table 4.1 ini.

Tabel 4.1 Hasil predeksi *training*

Matriks		Predict Class	
		Lahan Kosong	Bukan Lahan Kosong
Actual Class	Lahan Kosong	517	3
	Bukan Lahan	2	478

	Kosong		
--	--------	--	--

Dari tabel 4.1 bisa diketahui bahwa akurasi dari proses *training* sebesar 99.5%. Nilai akurasi ini bisa dinyatakan pada grafik hasil proses *training* setelah berjalan 1000 *epoch* menggunakan *tensorboard*.



Gambar 4.1 Grafik *accuracy* dan *loss*

4.2.3. Hasil Proses *Testing*

Proses *testing* menggunakan data uji sebanyak 300, dan setiap kelas jenis prediksi sebanyak 100 citra lahan kosong dan 200 citra bukan lahan kosong. Proses untuk mendapatkan akurasi dilakukan dengan 2 cara, yaitu menggunakan *method* `model.evaluate_generator()` dan dengan menggunakan perhitungan *Confusion matrix*.

Proses perhitungan akurasi bisa dilakukan langsung menggunakan *method* `model.evaluate_generator()` dengan cara memanggil *directory* dari folder validasi, kemudian dieksekusi dengan metode tersebut.

```

steps = nb_validation_samples/1

evaluate= model.evaluate_generator(validation_generator, steps)

print('Test accuracy = {:.2f}'.format(evaluate[1]*100)+'%')

```

Gambar 4.2 Pseudocode perhitungan akurasi

Hasil dari proses perhitungan akurasi menggunakan method `model.evaluate_generator()` ini, didapatkan nilai akurasi sebesar 95,63% dengan nilai *loss* 0,07. Dari proses perhitungan ini bisa diketahui bahwa hasil dari proses *testing* ini, lebih dari 90% data yang diuji coba hasilnya benar.

Sebagai penguat dari hasil perhitungan akurasi menggunakan `method.evaluate`, proses *testing* dilakukan dengan menguji semua data *testing* satu persatu. Hasil dari pengujian yang dilakukan bisa dilihat pada **Lampiran 2.**(Tabel Hasil uji Coba identifikasi lahan kosong) merupakan hasil identifikasi lahan kosong dari hasil cropping citra *google earth*. Kemudian dilanjutkan hasil uji coba identifikasi bukan lahan kosong dari hasil cropping citra *Google Earth*, dilihat pada **Lampiran 3.**(Tabel Hasil uji Coba identifikasi lahan bukan kosong)

Dari tabel-tabel pada lampiran 1 dan lampiran 2 dapat diketahui bahwa ada 5 data *image* yang salah pada kategori lahan kosong, 28 data *image* yang salah pada kategori bukan lahan kosong. Dalam mengidentifikasi jenis lahan kosong dan bukan lahan kosong yang terdapat pada *image* tersebut. Dari tabel lampiran dua dan lampiran tiga kemudian dimasukkan ke dalam tabel prediksi yang lebih ringkas seperti tabel 4.2 ini.

Tabel 4.2 Hasil prediksi proses testing

Matriks		Predict Class	
		Lahan Kosong	Bukan Lahan Kosong
Actual Class	Lahan Kosong	82	23
	Bukan Lahan Kosong	18	177

Berdasarkan tabel hasil prediksi tabel 4.2, hasil prediksi dari model terhadap data *testing* menunjukkan hasil yang cukup baik. Pada kelas lahan kosong dari 100 data yang diuji hanya ada 18 data yang memiliki prediksi salah. Sedangkan pada kelas bukan lahan kosong dari 200 data ada 23 data yang salah dari tabel tersebut bisa dihitung akurasi nya sebagai berikut

$$\text{Accuracy} = \frac{\text{All True Postive}}{\text{Total Number Testing Entries}} \times 100\%$$

$$\text{Accuracy} = \frac{259}{300} \times 100\% = 86,34\%$$

Dari perhitungan tersebut didapatkan nilai akurasi identifikasi lahan kosong berbasis citra *google earth* menggunakan algoritma CNN ini sebesar 86,34%. Hasil yang didapatkan dari perhitungan akurasi menggunakan *confusion matrix* ternyata lebih rendah dari pada perhitungan menggunakan *method*.

4.2.4. Pengaruh Parameter Learning Terhadap Akurasi

Salah satu hal yang mempengaruhi tinggi rendahnya akurasi yang didapatkan dalam proses pembentukan model adalah parameter *learning*. Dalam menentukan model terbaik, salah satu harus dicari nilai terbaik parameter *learning* dalam model CNN. Parameter *learning* yang dimaksud adalah pengaruh jumlah *epochs*, pengaruh nilai *learning rate* dan pengaruh ukuran *input* citra. Tujuan dari

penentuan parameter model ini ingin membandingkan model mana yang paling terbaik dengan memperhatikan nilai parameternya.

4.2.4.1. Pengaruh Jumlah *Epoch*

Dalam melakukan pembelajaran, ketika *dataset* sudah melalui proses *training* pada *neural network* sampai dikembalikan ke awal dalam satu putaran itulah yang disebut dengan *epoch*. Hingga saat ini belum ada penelitian yang melakukan klaim tentang jumlah *epoch* yang paling bagus dalam melakukan proses *training*. Namun jika merujuk pada penelitian yang dilakukan oleh Andre Lopez *at al.*, (2017), mereka melakukan percobaan antara 1000 hingga 2000 *xepoch*. Tapi disini mencoba untuk melakukan variasi percobaan lain seperti tabel berikut ini.

Tabel 4.3 Pengaruh *Epoch*

<i>Epocs</i>	Akurasi	<i>Loss value</i>	Waktu
500	83.67%	0.41	29 menit
1000	80.59%	0.76	41 menit
1500	85.25%	0.16	1 jam 6 menit

Berdasarkan tabel 4.3 dengan menggunakan nilai *learning rete* 0.001 dan dimensi 32x32 didapatkan akurasi yang paling bagus ada pada *epoch* 1500 yaitu 85,25%. Jika dilihat dari tabel tersebut, ketika *epoch* ditambah 500, akurasi bertambah dan nilai loss berkurang, dengan proses waktu juga bertambah lama.

4.2.4.2. Pengaruh Jumlah Nilai *Learning Rate*

Dalam melakukan pembelajaran dan *update* bobot pada *neuron* di *fully connected layer*, dibutuhkan metode optemasi seperti *Stochastic Gradient Descent* (SGD). Dengan menggunakan metode ini, proses *update* bobot bisa lebih optimal

dengan adanya parameter *learning* seperti nilai *learning rate*. Pada penelitian kali ini, salah satu menjadi analisis adalah pengaruh nilai *learning rate* terhadap akurasi yang didapatkan.

Tabel 4.4 Pengaruh nilai *learning rate*

<i>Learning rate</i>	Akurasi	<i>Loss value</i>	Waktu
0.01	74.54%	3.46	1 jam 5 menit
0.001	95.63%	0.07	1 jam 2 menit
0.0001	81.65%	0.16	59 menit

Jika dilihat dari tabel analisis 4.4, dengan jumlah *epochs* 1000 dan dimensi 32x32 perubahan nilai *learning* juga cukup mempengaruhi nilai akurasi dan *loss value*. Sedangkan untuk waktu eksekusi dengan jumlah *epochs* yang sama, perubahan waktu tidak terlalu signifikan. Berdasarkan tabel tersebut bisa diketahui bahwa nilai *learning rate* yang paling optimal adalah 0.001 dengan nilai akurasi 95,63% dan *value loss* sebesar 0,48.

4.2.4.3. Pengaruh Ukuran Citra

Salah satu parameter yang juga cukup mempengaruhi nilai akurasi adalah ukuran dari citra yang dijadikan masukan pada saat proses *training*. Karena semakin besar ukuran, otomatis semakin banyak nilai piksel yang perlu dilatih, hal itu membuat komputasi juga semakin meningkat. Selain itu, meningkatnya komputasi yang harus dilakukan membuat waktu pemrosesan juga semakin lama. Hal ini juga meningkatkan resiko terbentuknya model yang *overfitting*. Berikut pengaruh ukuran citra terhadap akurasi.

Tabel 4.5 Pengaruh ukuran citra

Ukuran Citra	Akurasi	<i>Loss value</i>	Waktu
32x32	95.63%	0.07	1 jam 5 menit
64x64	84.24%	1.42	1 jam 15 menit

128x128	86.98%	0.01	2 jam 3 menit
---------	--------	------	---------------

Proses pelatihan dijalankan dengan nilai *epoch* dan nilai *learning rate* 0.001. Dari percobaan tersebut dapat dilihat nilai akurasi malah menurun ketika meningkatkan ukuran dari *input-an*. Selain itu, dengan meningkatkan ukuran citra juga membuat waktu pelatihan juga semakin lama. Dari skenario pelatihan yang dijalankan ini bisa dilihat bahwa ukuran yang sesuai dengan arsitektur yang dibuat ada citra dengan ukuran 32x32.

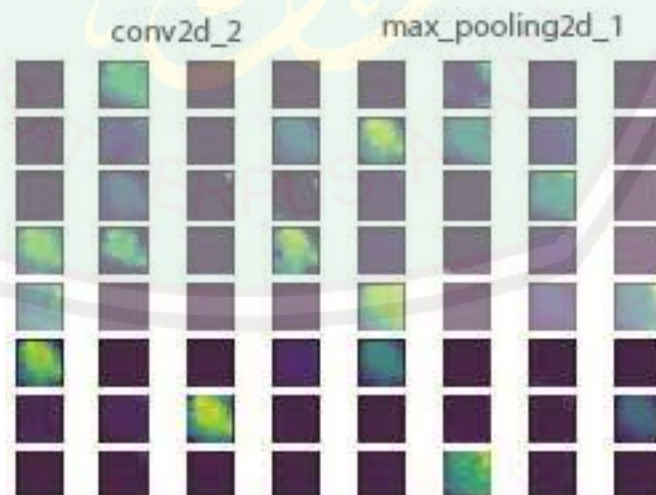
4.3. Pembahasan

Algoritma *Convolutional Neural Network* (CNN) merupakan salah satu algoritma *deep learning* yang fokus untuk mempelajari ciri dari suatu objek secara lebih mendalam. Salah satu yang menjadi bagian terpenting dari keberhasilan algoritma ini untuk melakukan klasifikasi terhadap suatu objek adalah banyaknya jumlah data. Semakin banyak jumlah data yang dilatih pada proses *training*, maka hasil yang didapatkannya pun akan semakin tinggi. Begitupun sebaliknya, algoritma CNN ini tidak akan optimal ketika data yang dilatih sedikit. Oleh karena itu, perlu adanya operasi ekstraksi tambahan untuk meningkatkan hasilnya.

Pada penelitian ini algoritma CNN berhasil untuk melakukan identifikasi lahan kosong kota Batu berbasis citra *google earth* dengan akurasi 95,63%. Jika melihat dari penelitian yang dijadikan rujukan yaitu penelitian yang dilakukan oleh Qinchuan Zhang *et al.*, (2016). Metode ini mengusulkan metode (CNN) untuk mendeteksi bangunan pinggiran kota secara otomatis dari citra Google Earth resolusi tinggi. Metode tradisional berdasarkan fitur rekayasa tangan tingkat rendah atau fitur tingkat menengah memiliki keterbatasan besar dalam lingkungan yang kompleks, terutama di daerah pinggiran kota. Terinspirasi oleh pencapaian

CNN yang luar biasa dalam pengenalan dan pendeteksian objek, penelitian mengembangkan metode baru untuk mendeteksi bangunan dalam gambar berantakan yang terdiri dari langkah-langkah sulit sebelumnya. Pertama, komputasi multi-skala yang digunakan untuk mengeksploitasi area bangunan dan pendekatan jendela geser diterapkan untuk menghasilkan daerah kandidat. Kemudian, CNN diterapkan untuk mengklasifikasikan wilayah. Akhirnya, peningkatan penindasan non maksimum digunakan untuk menghapus bangunan palsu. Peneliti menguji metode pada kumpulan gambar Google Earth yang sangat menantang dan mencapai presisi 89%, yang menunjukkan kekokohan dan efisiensi metode CNN.

Sebenarnya algoritma CNN ini merupakan algoritma yang sudah cukup bagus melakukan identifikasi lahan kosong berbasis citra *google earth*. Ditambah lagi dengan kemandirian algoritma yang sebenarnya sudah tidak membutuhkan ekstraksi fitur lagi, karena didalamnya sudah ada proses *feature learning*.



Gambar 4.3 Hasil *feature learning* tahap 1

Walapun begitu karena komputasinya yang terlalu tinggi membuat algoritma *overfitting* sehingga akurasi yang didapatkan tidak setinggi yang diharapkan. Oleh karena itu, peneliti menggunakan image RGB agar informasi yang didapatkan oleh model bertambah banyak sehingga akurasi yang didapatkan dari model CNN akan tinggi.

Selain masalah *overfitting*, masalah yang membuat hasil dari klasifikasi ini kurang maksimal adalah terbatasnya kemampuan perangkat digunakan, dan di masa pandemi virus covid-19 ini ruang gerak menjadi terbatas untuk melakukan survey lokasi lahan kosong kota Batu. Hal itu membuat tidak mampu bereksperimen lebih untuk meningkatkan hasil dari penelitian ini.

Melihat dari pembahasan penelitian bisa dilihat bahwa pentingnya untuk belajar dan mempelajari sesuatu secara mendalam. Ketika algoritma tidak mampu mempelajari suatu objek secara lebih dalam, maka hasil yang didapatkannya pun akan berkurang. Hal ini tidak beda dengan manusia, ketika kita tidak mau untuk belajar dan mempelajari sesuatu maka *output* dari diri kita pasti kurang baik.

Proses untuk selalu belajar dan mempelajari sesuatu ini selaras dengan wahyu pertama yang Allah turunkan kepada nabi Muhammad ﷺ yaitu QS al-‘Alaq/96: 1-5

إِقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ (١) خَلَقَ الْإِنْسَانَ مِنْ عَلَقٍ (٢) إِقْرَأْ وَرَبُّكَ الْأَكْرَمُ (٣) الَّذِي
عَلَّمَ بِالْقَلَمِ (٤) عَلَّمَ الْإِنْسَانَ مَا لَمْ يَعْلَمْ

“(Bacalah) maksudnya mulailah membaca dan memulainya (dengan menyebut nama Rabbmu yang menciptakan semua makhluk). (Dia telah menciptakan manusia) atau jenis manusia (dari ‘alaq) lafal ‘Alaq bentuk jamak dari lafal ‘Alaqah, artinya segumpal darah yang kental). (Bacalah) lafal ayat ini

mengukuhkan makna lafal pertama yang sama (dan Rabbmulah Yang Paling Pemurah) artinya tiada seorang pun yang dapat menandingi kemurahan-Nya. Lafal ayat ini dalam lafal Iqra'. (Yang mengajar) manusia (apa yang tidak diketahuinya) yaitu sebelum Dia mengajarkan kepadanya hidayah, menulis dan berkreasi serta hal hal lainnya”(QS al-‘Alaq/96: 1-5).

QS al-‘Alaq/96: 1-5, mengandung pesan ontologis tentang belajar dan pembelajaran. Dalam hal ini, nabi Muhammad ﷺ, yang ummi (buta huruf aksara) beliau diperintahkan untuk tetap belajar. Hasil yang ditimbulkan dengan usaha belajar tersebut, dapat menghasilkan ilmu agama seperti fikih, tauhid, akhlaq dan sebagainya yang Allah tranferkan melalui lisan beliau. Sedangkan hasil yang ditimbulkan dengan usaha membaca ayat-ayat *kawniyah*, dapat menghasilkan sains seperti fisika, biologi, kimia, astronomi, dan sebagainya. Dapat dirumuskan bahwa ilmu yang bersumber dari ayat-ayat *qur’aniyah* dan *kawniyah*, harus diperoleh melalui proses belajar membaca.

Timbul pertanyaan, mengapa kata *iqra’* atau perintah membaca sederetan ayat QS al-‘Alaq/96: 1-5 terulang dua kali yakni pada ayat satu dan tiga. Jika merujuk pada..kitab tafsir ibnu katsir rahimahullah, beliau berkata Seseorang itu akan semakin mulia dengan ilmu diin yang ia miliki. Ilmu itulah yang membedakan bapak manusia, yaitu Adam dengan para malaikat. Ilmu ini terkadang di pikiran. Ilmu juga kadang di lisan. Ilmu juga terkadang di dalam tulisan tangan untuk menyalurkan apa yang dalam pikiran, lisan, maupun yang tergambar di pikiran.

Bisa disimpulkan bahwa kemampuan manusia untuk belajar dan mempelajari inilah yang membuat manusia mulia disisi Allah SWT. Selain itu

Allah juga mengisyaratkan bahwa Allah menciptakan tubuh kita ini agar dimanfaatkan untuk belajar. Hal ini juga diisyaratkan Allah pada QS An-Nahl(16):78 berbicara tentang komponen pada diri manusia yang harus digunakan dalam kegiatan belajar dan pembelajaran:

وَاللَّهُ أَخْرَجَكُمْ مِنْ بُطُونِ أُمَّهَاتِكُمْ لَتَعْلَمُونَ شَيْئًا وَجَعَلَ لَكُمُ السَّمْعَ وَالْأَبْصَارَ وَالْأَفْئِدَةَ لَعَلَّكُمْ

تَشْكُرُونَ (٨٧)

“(Dan Allah mengeluarkan kalian dari perut ibu kalian dalam keadaan tidak mengetahui sesuatu pun) jumlah kalimat Laa ta’muuna syaian berkedudukan menjadi hal atau kalimat keterangan (dan Dia memberi kalian pendengaran) lafal as-sam’u bermakna jamak sejakali pun lafalnya mufrad(penglihatan dan hati) kalbu (agar kalian bersyukur) kepada-Nya atas hal – hal tersebut, oleh karenanya kalian beriman kepada-Nya”(QS An-Nahl(16):78).

Ayat pada QS An-Nahl(16):78 mengisyarat adanya tiga komponen adanya tiga komponen yang terlibat dalam teori pembelajaran, yaitu: al-sam’a, al-bashar dan al-fu’ad. Secara leksial, kata al-sam’a berarti telinga yang fungsinya menangkap suara, memahami pembicaraan, dan selainnya. Penyebutan al-sama dalam Al-qur’an seringkali dihubungkan dengan penglihatan dan qalbu, yang menunjukkan adanya saling melengkapi antara berbagai alat itu dalam kegiatan belajar dan mempelajari sesuatu. Tidak sekedar akal pikiran saja, dalam mempelajari sesuatu manusia juga perlu menggunakan berbagai komponen yang ada dalam tubuhnya agar mampu mempelajari sesuatu secara lebih mendalam.

Kemudian dilanjutkan lagi pada QS An-Nahl (16): 79 untuk mengamati kebesaran-kebesaran Allah SWT

أَلَمْ يَرَوْا إِلَى الطَّيْرِ مُسَخَّرَاتٍ فِي جَوْ السَّمَاءِ مَا يُمَسِّكُهُنَّ إِلَّا اللَّهُ إِنَّ فِي ذَلِكَ
لَآيَاتٍ لِّقَوْمٍ يُؤْمِنُونَ (٧٩)

“(Tidakkan mereka memperhatikan burung-burung yang dimudahkan) terbang (di angkasa bebas) di udara antara langit dan bumi. (tidak ada yang menahannya) sewaktu ia melipat sayap atau mengembangkannya sehingga ia tidak jatuh ke bawah (selain daripada Allah) yakni dengan kekuasaan-Nya. (sesungguhnya pada yang demikian itu benar-benar terdapat tanda-tanda kekuasaan Allah bagi orang-orang yang beriman) yaitu penciptaan burung itu sehingga dapat memungkinkan bagi burung untuk terbang mengaranginya dan menahan burung untuk jatuh ke tanah”(QS An-Nahl(16):79).

Setelah Allah mengisyaratkan bahwa Allah telah memberi pendengaran, penglihatan dan hati dengan suatu tujuan tertentu. Allah memerintahkan umat manusia untuk memperhatikan berbagai hal tentang penciptaan-Nya. Dicontohkan dalam ayat tersebut adalah perintah untuk mengamati burung-burung yang terbang diangkasa. Secara eksplisit Allah memerintahkan manusia untuk berfikir bahwa segala sesuatu yang terjadi di dunia, bisa terjadi karena kekuasaan Allah SWT.

BAB V

PENUTUP

5.1. Kesimpulan

Dari hasil penelitian yang dikerjakan untuk mengidentifikasi lahan kosong berbasis citra *google earth* menggunakan metode *Convolutional Neural Network* (CNN) ini, dapat ditarik kesimpulan sebagai berikut:

Algoritma CNN sudah cukup baik dalam melakukan identifikasi lahan kosong berbasis citra *google earth* ini. Dengan melakukan *training* pada 1000 *epoch* dan nilai *learning rate* 0.001 didapatkan akurasi *training* sebesar 99,5% dan akurasi *testing* sebesar 86,34%. Hasil ini sudah cukup bagus melihat kualitas gambar dari hasil cropping citra *google earth* dan jumlah data yang didapatkan tidak begitu bagus dan jumlahnya data yang terdiri dari 1300 *image* yaitu 1000 *image* data *training* dan 300 *image* data *testing*.

5.2. Saran

Dalam pengembangan sistem identifikasi lahan kosong berbasis citra *google earth* menggunakan algoritma CNN ini diperlukan beberapa perbaikan untuk mencapai hasil yang lebih maksimal, maka perlu dilakukan perbaikan pada beberapa hal diantaranya:

1. Memperbaiki kualitas citra hasil cropping citra *google earth* yang *terupgrade*, selain itu memperbesar dimensi citra dan memperbanyak jumlah data agar algoritma mampu bekerja lebih maksimal.
2. Menambah jenis variable kelas yang diidentifikasi agar kemampuan algoritma CNN lebih terukur.

DAFTAR PUSTAKA

- Albani, Youssef, & Suriani. (2017). A Deep Learning Approach for Object Recognition with NAO Soccer Robots. *Sapienza University of Rome*.
- Abhirawa, Halprin., et.al. *Pengenalan Wajah Menggunakan Convolutional Neural Network. E-Proceeding of Engineering: Vol.4, No.3 Desember 2017 ISSN 2355-9365*
- Andre Lopes, E. d. (2016). *Facial Expression Recognition with Convolutional Neural Network: Coping with Few Data and the Training Sample Order*
- Arifqoh, & Harintaka (2018). *Implementasi Metode Convolutional Neural Network untuk Klasifikasi Tanaman pada citra Resolusi tinggi. Program Studi Teknik Geomatika, Departemen Teknik Geodesi, Fakultas Teknik, Universitas Gadjah Mada*
- Ariastita, Putu Dde (2005). *Konsep Penanganan Lahan Kosong dalam Rangka Pengembangan Lahan Kosong Perkotaan (konsep pengembangan Lahan Kosong Berbasis Stakeholders). conference: Seminar Nasional ASPI Untar. Universitas Taruma Negara*
- Chapin Jr Stuart & Edward J Kaiser (1979), *Urban Land Use Planning*, University of Illinois Press
- Danukusumo, Kefin Pudi. 2017. *Implementasi Deep Learning Menggunakan Convolutional Neural Network untuk Klasifikasi Citra Candi Berbasis GPU. Tugas Akhir. Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Atma Jaya Yogyakarta.*
- Dewi, Syarifah Rosita. 2018. *Deep Learning Object Detection pada Video Menggunakan Tensorflow dan Convolutional Neural Network. Tugas Akhir. Program Studi Statistika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Islam Indonesia Yogyakarta*
- Deng, L., & Yu, D. (2013). *Deep Learning: Methods and Applications. Foundations and Trends® in Signal Processing*, 7 (3–4), 197–387
- Dosovitskiy, Alexey., et.al. *Learning to Generate Chairs, Tables and Cars with Convolutional Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence.*
- Dertat, Arden. *Applied Deep Learning - Part 4: Convolutional Neural Networks. Towardsdatascience.*
- Emilio Guirado, Siham Tabik, Domingo Alcaraz-Segura, Javier Cabello, and Francisco Herrera. 2017. *Deep-Learning Convolutional Neural Network*

for scattered shrub detection with Google Earth Imagery. Andalusia Center for Assesment and monitoring of global change (CAESGC).

- Fausett, Laurene. 1994. *Fundamental of Neural Network : Architecture, Algorithm, and Application.* New Jersey : Prentice-Hall.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning (Adaptive Computation and Mechine Learning Series).* The IMT Press.
- Hermawan. (2006). *Jaringan Syaraf Tiruan dan Aplikasinya.* Yogyakarta: Andi.
- Hubel, & Wiesel. (1961). *Receptive Fields, Binocular Interaction And Functional Architecture In The Cat's Visual Cortex.* Harvard Aledical School.
- Irwan, B.S.(2015). *Deteksi Non-RTH(Ruang Terbuka Hijau) Kota Malang Berbasis Citra Google Earth Dengan Menggunakan Naïve Bayes Classifie.* Faculty of Science And Technology, Maulana Malik Ibrahim State Islamic University, Malang.
- Iriyanto dan Zaini. 2014. *Pengolahan kosong Citra Digital.* Bandar Lampung : Anugerah Utama Raharja Printing dan Publishing
- Kivell, Phillep (1993). *Land and The City Patterns and Process of Urban Change.* Routledge, New York
- Khaeriyah, Rakhil. 2019. *Implementasi Metode Convolutional Neural Network Menggunakan Tensorflow Dalam Mendeteksi Sebuah Objek.* Faku;tas Matematika dan Ilmu Pengetahuan Alam Universitas Indonesia.
- Kriesel, D. (2005). *A Brief Introduction to Neural Networks.* drkriesel.com.
- Krizhevsky, A., Sutskever, I., & Hinton, G. (2012). *ImageNet Classification with Deep Convolutional Neural Networks.* University of Toronto.
- Li, H., Wang, P., You, M., & Shen, C. (2018). Reading car license plates using deep neural networks. *Elsevier.*
- Liu, Tianyi, dkk. 2015. *Implementation of Training Convolutional Neural Networks.* <https://arxiv.org/ftp/arxiv/papers/1506/1506.01195.pdf>. Diakses pada 6 Februari 2018
- Miracle, Budi, Sutrisno (2018). *Implementasi Metode Backpropagation untuk Prediksi Harga Batu Bara.* Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 6502
- Mubarok Hamdani. 2017. *Identifikasi Ekspresi Wajah Berbasis Citra Menggnakan Algortima Convolutional Neural Network(CNN).* Jurnal MATICS, No. 1, Vol. 9.

- Muhammad Jaleed Khan, Adef Yousaf, Nizwa Javed, Shifa Nadeem, and Khurram Khurshid. 2017. *Automatic Target Detection in Satellite Images using Deep Learning*. Journal of Space Technology, Vol 7, No1.
- Murni, Aniati dan Suryana Setiawan. 1992. *Pengantar Pengolahan kosong Citra Digital*. Jakarta : Alex Media Komputindo.
- Novyantika, Rizky Dwi. 2018. *Deteksi Nomor Kendaraan Bermotor Pada Media Streaming dengan Algoritma Convolutional Neural Network Menggunakan Tensorflow*. Tugas Akhir. Program Studi Statistika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Islam Indonesia Yogyakarta
- Peraturan Pemerintah No.36 Tahun 1998 tentang *Penertiban dan Pendayagunaan Lahan Terlantar*
- Peraturan Menteri Agraria/Kepala BPN No.3 Tahun 1998 *tentang Pemanfaatan Lahan Kosong untuk Tanaman Pangan*
- Putra, Darma. 2010. *Pengolahan Citra Digital*. Yogyakarta : Andi Offset.
- Rahman, Nouroz. 2017. *What is the benefit of using average pooling rather than max pooling*. <https://www.quora.com/What-is-the-benefit-of-using-averagepooling-rather-than-max-pooling>. Diakses pada tanggal 8 Februari 2018.
- Sena, s. (2017, November 13). Convolutional Neural Network. Retrieved from medium.com: <https://medium.com/@samuelsena/pengenalan-deep-learning-part-7-convolutional-neural-network-cnn-b003b477dc94/>
- Sharma, Sagar. 2017. *Activation Functions : Neural Networks*. <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>. Diakses pada 8 Februari 2018
- Setiyono, Budi dan Zufar, Muhammad. 2016. *Convolutional Neural Network untuk Pengenalan Wajah Secara Real-Time*. Jurnal Sains dan Seni ITS Vol. 5 No.2 (2016) 2337-3520
- Taufiq, Imam. 2018. *Deep Learning Untuk Deteksi Tanda Nomor Kendaraan Bermotor Menggunakan Algoritma Convolutional Neural Network Dengan Python Dan Tensorflow*. Skripsi. Program Studi Sistem Informasi Sekolah Tinggi Manajemen Informatika dan Komputer AKAKOM.
- Qinbin Hou, Ming-Ming Cheng, Xiaowei Hu, Ali Borji, Zhuowen Tu, Philip H. S. Torr. 2018. *Deeply Supervised Salient Object Detection with Short Connections*. IEEE Transactions On Pattern Analysis and Machine Learning

Qinchuan Zhang, Yunhong Wang, Qingjie Liu, Xiangyu Liu, Wei Wang. 2017. *CNN Based Suburban Building Detection Using Monocular, High Resolution Google Earth Image*. IEEE, 2017, 978-1-5090-3332-4

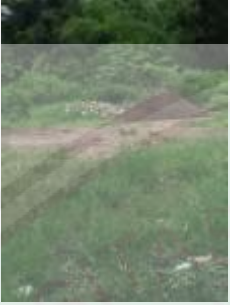




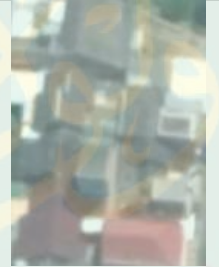

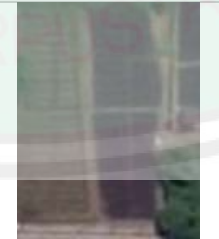
X. Ouyang, P. Zhou, C. H. Li, and L. Liu, "Sentiment Analysis Using Convolutional Neural Network," 2015 IEEE Int. Conf. Comput. Inf. Technol. Ubiquitous Comput. Commun. Dependable, Auton. Secur. Comput. Pervasive Intell. Comput., 2015, pp. 2359–2364.

Zhiling Guo, Xiaowai Shao, Yongwei Xu, Hiroyuki Miyazaki, Wataru Ohira and Ryosuke Shibasaki. 2016. *Identificaation of Village Building via Google Earth Images and Supervised Machine Learning Methos*. Center For Spatial Information Science, University of Tokyo.



LAMPIRAN-LAMPIRAN

Lampiran 1. Perbandingan data (Tabel Perbandingan Data Lahan Sebenarnya dengan hasil *Cropping Citra Google Earth*)

Jenis Lahan	Lahan Sebenarnya	Citra <i>Google Earth</i>	Tahun lahan sebenarnya	Tahun Citra <i>Google Earth</i>
Lahan Kosong			Foto diambil pada Januari 2020	2020 Google Image © Maxar Technologies
Lahan Hijau			Foto diambil pada Januari 2020	2020 Google Image © Maxar Technologies
Lahan Perumahan			Foto diambil pada Januari 2020	2020 Google Image © Maxar Technologies
Lahan Sawah			Foto diambil pada Februari 2020	2020 Google Image © Maxar Technologies

Lampiran 2. Hasil pengujian (Tabel Hasil uji Coba identifikasi lahan kosong)

No	Input	Output	Status Identifikasi
1	lahan_kosong.jpg	lahan kosong	Benar
2	lahan_kosong.jpg	lahan kosong	Benar
3	lahan_kosong.jpg	lahan kosong	Benar
4	lahan_kosong.jpg	lahan kosong	Benar
5	lahan_kosong.jpg	lahan kosong	Benar
6	lahan_kosong.jpg	lahan kosong	Benar
7	lahan_kosong.jpg	lahan kosong	Benar
8	lahan_kosong.jpg	lahan kosong	Benar
9	lahan_kosong.jpg	lahan kosong	Benar
10	lahan_kosong.jpg	lahan kosong	Benar
11	lahan_kosong.jpg	lahan kosong	Benar
12	lahan_kosong.jpg	lahan kosong	Benar
13	lahan_kosong.jpg	lahan kosong	Benar
14	lahan_kosong.jpg	lahan kosong	Benar
15	lahan_kosong.jpg	lahan kosong	Benar
16	lahan_kosong.jpg	lahan kosong	Benar
17	lahan_kosong.jpg	lahan kosong	Benar
18	lahan_kosong.jpg	lahan kosong	Benar
19	lahan_kosong.jpg	lahan kosong	Benar
20	lahan_kosong.jpg	lahan kosong	Benar
21	lahan_kosong.jpg	lahan kosong	Benar
22	lahan_kosong.jpg	lahan kosong	Benar
23	lahan_kosong.jpg	lahan kosong	Benar
24	lahan_kosong.jpg	lahan kosong	Benar
25	lahan_kosong.jpg	lahan kosong	Benar
26	lahan_kosong.jpg	lahan kosong	Benar
27	lahan_kosong.jpg	lahan kosong	Benar
28	lahan_kosong.jpg	lahan kosong	Benar
29	lahan_kosong.jpg	lahan kosong	Benar
30	lahan_kosong.jpg	lahan kosong	Benar
31	lahan_kosong.jpg	lahan kosong	Benar
32	lahan_kosong.jpg	lahan kosong	Benar
33	lahan_kosong.jpg	lahan kosong	Benar
34	lahan_kosong.jpg	lahan kosong	Benar
35	lahan_kosong.jpg	lahan kosong	Benar
36	lahan_kosong.jpg	lahan kosong	Benar
37	lahan_kosong.jpg	lahan kosong	Benar
38	lahan_kosong.jpg	lahan kosong	Benar
39	lahan_kosong.jpg	lahan kosong	Benar
40	lahan_kosong.jpg	lahan kosong	Benar

41	lahan_kosong.jpg	lahan kosong	Benar
42	lahan_kosong.jpg	lahan kosong	Benar
43	lahan_kosong.jpg	bukan lahan kosong	Salah
44	lahan_kosong.jpg	lahan kosong	Benar
45	lahan_kosong.jpg	lahan kosong	Benar
46	lahan_kosong.jpg	bukan lahan kosong	Salah
47	lahan_kosong.jpg	lahan kosong	Benar
48	lahan_kosong.jpg	lahan kosong	Benar
49	lahan_kosong.jpg	lahan kosong	Benar
50	lahan_kosong.jpg	lahan kosong	Benar
51	lahan_kosong.jpg	lahan kosong	Benar
52	lahan_kosong.jpg	lahan kosong	Benar
53	lahan_kosong.jpg	lahan kosong	Benar
54	lahan_kosong.jpg	lahan kosong	Benar
55	lahan_kosong.jpg	lahan kosong	Benar
56	lahan_kosong.jpg	lahan kosong	Benar
57	lahan_kosong.jpg	lahan kosong	Benar
58	lahan_kosong.jpg	lahan kosong	Benar
59	lahan_kosong.jpg	lahan kosong	Benar
60	lahan_kosong.jpg	lahan kosong	Benar
61	lahan_kosong.jpg	lahan kosong	Benar
62	lahan_kosong.jpg	lahan kosong	Benar
63	lahan_kosong.jpg	lahan kosong	Benar
64	lahan_kosong.jpg	lahan kosong	Benar
65	lahan_kosong.jpg	lahan kosong	Benar
66	lahan_kosong.jpg	lahan kosong	Benar
67	lahan_kosong.jpg	lahan kosong	Benar
68	lahan_kosong.jpg	lahan kosong	Benar
69	lahan_kosong.jpg	lahan kosong	Benar
70	lahan_kosong.jpg	lahan kosong	Benar
71	lahan_kosong.jpg	bukan lahan kosong	Salah
72	lahan_kosong.jpg	bukan lahan kosong	Salah
73	lahan_kosong.jpg	bukan lahan kosong	Salah
74	lahan_kosong.jpg	lahan kosong	Benar
75	lahan_kosong.jpg	bukan lahan kosong	Salah
76	lahan_kosong.jpg	lahan kosong	Benar
77	lahan_kosong.jpg	bukan lahan kosong	Salah
78	lahan_kosong.jpg	bukan lahan kosong	Salah
79	lahan_kosong.jpg	lahan kosong	Benar
80	lahan_kosong.jpg	lahan kosong	Benar
81	lahan_kosong.jpg	lahan kosong	Benar
82	lahan_kosong.jpg	lahan kosong	Benar
83	lahan_kosong.jpg	lahan kosong	Benar

84	lahan_kosong.jpg	lahan kosong	Benar
85	lahan_kosong.jpg	bukan lahan kosong	Salah
86	lahan_kosong.jpg	lahan kosong	Benar
87	lahan_kosong.jpg	bukan lahan kosong	Salah
88	lahan_kosong.jpg	bukan lahan kosong	Salah
89	lahan_kosong.jpg	lahan kosong	Benar
90	lahan_kosong.jpg	lahan kosong	Benar
91	lahan_kosong.jpg	lahan kosong	Benar
92	lahan_kosong.jpg	lahan kosong	Benar
93	lahan_kosong.jpg	bukan lahan kosong	Salah
94	lahan_kosong.jpg	bukan lahan kosong	Salah
95	lahan_kosong.jpg	bukan lahan kosong	Salah
96	lahan_kosong.jpg	bukan lahan kosong	Salah
97	lahan_kosong.jpg	bukan lahan kosong	Salah
98	lahan_kosong.jpg	bukan lahan kosong	Salah
99	lahan_kosong.jpg	bukan lahan kosong	Salah
100	lahan_kosong.jpg	lahan kosong	Benar

Lampiran 3. Hasil pengujian (Tabel Hasil uji Coba identifikasi bukan lahan kosong)

No	Input	Output	Status Identifikasi
1	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
2	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
3	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
4	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
5	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
6	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
7	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
8	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
9	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
10	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
11	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
12	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
13	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
14	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
15	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
16	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
17	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
18	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
19	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar

20	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
21	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
22	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
23	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
24	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
25	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
26	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
27	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
28	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
29	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
30	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
31	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
32	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
33	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
34	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
35	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
36	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
37	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
38	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
39	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
40	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
41	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
42	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
43	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
44	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
45	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
46	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
47	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
48	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
49	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
50	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
51	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
52	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
53	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
54	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
55	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
56	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
57	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
58	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
59	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
60	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
61	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
62	Bukan Lahan Kosong.jpg	lahan_kosong	Salah

63	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
64	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
65	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
66	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
67	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
68	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
69	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
70	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
71	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
72	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
73	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
74	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
75	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
76	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
77	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
78	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
79	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
80	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
81	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
82	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
83	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
84	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
85	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
86	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
87	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
88	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
89	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
90	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
91	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
92	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
93	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
94	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
95	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
96	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
97	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
98	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
99	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
100	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
101	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
102	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
103	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
104	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
105	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar

106	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
107	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
108	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
109	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
110	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
111	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
112	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
113	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
114	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
115	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
116	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
117	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
118	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
119	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
120	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
121	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
122	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
123	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
124	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
125	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
126	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
127	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
128	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
129	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
130	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
131	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
132	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
133	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
134	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
135	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
136	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
137	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
138	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
139	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
140	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
141	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
142	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
143	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
144	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
145	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
146	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
147	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
148	Bukan Lahan Kosong.jpg	lahan_kosong	Salah

149	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
150	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
151	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
152	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
153	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
154	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
155	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
156	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
157	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
158	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
159	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
160	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
161	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
162	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
163	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
164	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
165	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
166	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
167	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
168	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
169	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
170	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
171	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
172	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
173	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
174	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
175	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
176	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
177	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
178	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
179	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
180	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
181	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
182	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
183	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
184	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
185	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
186	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
187	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
188	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
189	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
190	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
191	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar

192	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
193	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
194	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
195	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
196	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
197	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
198	Bukan Lahan Kosong.jpg	lahan_kosong	Salah
199	Bukan Lahan Kosong.jpg	bukan_lahan_kosong	Benar
200	Bukan Lahan Kosong.jpg	lahan_kosong	Salah

