

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Penelitian-Penelitian terkait

Beberapa penelitian dalam mendeteksi dan mengklasifikasi kerusakan jalan telah dilakukan, baik dengan metode *image processing*, *machine learning* maupun *deep learning*. Presisi dan akurasi dari proses deteksi yang dilakukan juga memberikan hasil yang baik.

Salah satu penelitian yang mengenai pendeteksian kerusakan jalan yang menunjukkan hasil yang baik adalah penelitian dengan judul "*Image-Based Pothole Detection System for ITS Service and Road Management System*" oleh Ryu pada tahun 2015. Penelitian ini membahas tentang deteksi lubang pada jalan melalui beberapa tahap segmentasi, yang mencakup binarisasi dengan *HST*, median *filter*, *closing*, dan ekstraksi kandidat (ukuran, kekompakan, *keeliptisan*, dan *kelinieran*). Selanjutnya, lubang ditentukan menggunakan metode *OHI*. Metode ini berhasil mencapai tingkat presisi sebesar 80%.

Pendeteksian lubang juga telah dilakukan pada penelitian dengan judul *A Novel Technique for Automatic Road Distress Detection and Analysis*, (Daniel, 2014). Penelitian ini membahas tentang pendeteksian citra lubang dan retak. Beberapa tahap pengolahan citra dan *Extract feature* yang digunakan yaitu *grayscale*, *filtering*, *edge detection* dan *connected component labelling*. Sedangkan metode klasifikasi yang digunakan adalah *Support Machine Learning*.

Penelitian terkait pengidentifikasian kerusakan jalan juga pernah diteliti oleh (Mandaya, 2020) dengan judul pemanfaatan teknologi *UAV (Unmanned Aerial Vehicle)* untuk identifikasi dan klasifikasi jenis-jenis kerusakan jalan. Pada penelitian ini algoritma yang digunakan untuk mendeteksi dan mengklasifikasi kerusakan jalan adalah dengan menggunakan algoritma *SIFT*, dengan kerusakan jalan yang digunakan adalah kerusakan jalan jenis lubang, retak kulit buaya, tambalan, amblas, sungkur dan retak tepi dengan hasil *overall accuracy* sebesar 96,36% dan nilai *Kappa* sebesar 94,15%.

Penelitian lain mengenai deteksi kerusakan jalan dengan metode *CNN* juga pernah diteliti oleh (Fajar, dkk, 2022) dengan judul Identifikasi Kerusakan Jalan dengan Metode *Faster R-CNN* Studi Kasus di Jalan Pakansari Bogor Jawa Barat. Penelitian ini membahas mengenai hasil akurasi dan kecepatan deteksi dari model yang telah ada sebelumnya, dengan melakukan perubahan pada algoritma pengumpulan region proposal pada *Fast R-CNN* yang semula selektif *search* menjadi *Region Proposal Network (RPN)*. Metode tersebut yaitu *Faster R-CNN*. Masing-masing kecepatan *R-CNN*, *Fast R-CNN*, *Faster R-CNN* berturut-turut yaitu 50 detik, 2 detik, dan 0,2 detik. Dengan catatan, *hardware* yang digunakan pada penelitian tersebut menggunakan komputer *high-end*. Pada pengukuran akurasi, keluarga *R-CNN* mendapat akurasi rata-rata diatas 70%. Dengan metode *Faster R-CNN* yang telah diperbarui, hasil akurasi yang diperoleh yaitu sebesar 96,84% sedangkan kecepatan deteksi yang diperoleh sebesar 0,16 citra/detik.

## 2.2 Pengertian jalan

Dalam setiap kegiatan masyarakat tentunya tidak terlepas dari proses perpindahan dan memindahkan baik berupa barang maupun jasa. Proses pendistribusian atau perpindahan ini akan sangat erat kaitanya dengan perkembangan ekonomi yang ada pada wilayah tersebut, semakin cepat proses pendistribusian maka tingkat perekonomian masyarakat juga akan berdampak baik. Selain itu dengan kondisi jalan yang baik maka akan dapat menurunkan tingkat kecelakaan dalam berlalu lintas.

Jalan adalah prasarana transportasi darat yang meliputi segala bagian jalan, termasuk bangunan pelengkap dan perlengkapannya yang diperuntukan bagi lalu lintas, yang berada pada permukaan tanah, di atas permukaan tanah, di bawah permukaan tanah atau air, seta di atas permukaan air, kecuali jalan kereta api, dan jalan kabel (Undang-Undang Republik Indonesia No.2/2022, pasal 1 ayat 1). Jalan merupakan salah satu unsur penting dalam pengembangan kehidupan berbangsa dan bernegara serta memiliki peranan penting dalam memberikan dukungan dibidang ekonomi, sosial dan budaya. Sesuai dengan peranannya, jalan akan selalu berhubungan penting dengan kehidupan masyarakatnya.

Berdasarkan Undang-Undang, jalan terbagi menjadi 3 jenis yaitu ;

a. Jalan Umum

Merupakan jenis jalan yang digunakan dalam proses operasionalnya diperuntukan untuk lalu lintas umum.

b. Jalan Khusus

Merupakan jenis Jalan yang dibangun oleh Badan Usaha, Instansi, perseorangan, ataupun kelompok masyarakat yang membangun jalan untuk kepentingan pribadinya.

c. Jalan Tol

Merupakan jenis jalan umum yang masih merupakan bagian dari sistem jaringan jalan serta sebagai jalan nasional yang mana dalam pengoperasiannya pengguna wajib membayar pembayaran tol.

### 2.3 Klasifikasi Jalan

Berdasarkan medan jalan, jenis klasifikasi jalan dapat dibedakan menjadi 3 kategori yaitu jenis medan Datar, Berbukit, dan Pegunungan. Proses pengklasifikasian berdasarkan medan jalan dilakukan dengan memperhatikan kondisi kemiringan medan yang diukur secara tegak lurus terhadap garis kontur. Keseragaman dari kondisi medan yang akan diproyeksi harus mempertimbangkan keseragaman dari kondisi medan yang sudah tersusun menurut rencana trase dengan cara mengabaikan perubahan-perubahan pada bagian kecil dari segmen rencana (Departemen Pekerjaan Umum Direktorat Jenderal Bina Marga, 1997).

Tabel 2. 1 Klasifikasi Jalan Berdasarkan Medan Jalan (Departemen Pekerjaan Umum Direktorat Jenderal Bina Marga, 1997)

| No. | Jenis Medan | Notasi | Kemiringan Medan (%) |
|-----|-------------|--------|----------------------|
| 1   | Datar       | D      | < 3                  |
| 2   | Berbukit    | B      | 3-25                 |
| 3   | Pegunungan  | G      | > 25                 |

Sedangkan berdasarkan (Undang-Undang Republik Indonesia No. 2/2022) tentang lalu lintas dan angkutan jalan didefinisikan jalan adalah seluruh bagian jalan, termasuk bangunan pelengkap dan perlengkapannya yang diperuntukan bagi lalu lintas umum, yang berada pada permukaan tanah, di atas permukaan tanah, di bawah permukaan tanah dan air, serta di atas permukaan air, kecuali Jalan Rel dan Jalan Kabel. Menurut statusnya jalan terbagi menjadi 5 jenis yaitu :

1. Jalan Nasional adalah jalan yang menghubungkan provinsi (antar provinsi). Jalan nasional terdiri atas jalan arteri primer, jalan kolektor primer yang menghubungkan antar ibukota provinsi, jalan tol, dan jalan strategis nasional.
2. Jalan Provinsi adalah sistem jalan yang menghubungkan antar kabupaten/kota di dalam satu provinsi. Jalan provinsi terdiri dari beberapa jenis, termasuk jalan kolektor primer yang menghubungkan ibukota provinsi dengan ibukota kabupaten/kota, jalan strategis provinsi (kecuali jalan arteri primer), jalan kolektor primer yang menghubungkan antar ibukota provinsi, jalan tol dan jalan strategis nasional.
3. Jalan kabupaten adalah jalan yang menghubungkan antar kelurahan/desa disebut jalan kabupaten. Jalan kabupaten terdiri dari beberapa jenis, termasuk jalan kolektor primer yang bukan termasuk jalan nasional dan jalan, jalan lokal primer yang menghubungkan ibukota kabupaten dengan ibukota kecamatan, ibukota kabupaten dengan pusat desa, antar ibukota kecamatan, ibukota kecamatan dengan desa. Selain itu juga terdapat jalan sekunder yang tidak termasuk jalan provinsi dan jalan sekunder dalam kota, serta jalan strategis kabupaten.
4. Jalan kota adalah jalan umum yang menghubungkan antar pusat pelayanan dalam kota, menghubungkan pusat pelayanan dengan persil, menghubungkan antara persil, serta menghubungkan antar pusat permukiman yang berada di dalam kota.
5. Jalan desa merupakan jalan umum yang menghubungkan kawasan dan/atau antar permukiman di dalam desa, serta jalan lingkungan.

### 2.3.1 Jenis Kerusakan Perkerasan Jalan Lentur

Menurut Kementrian Pekerjaan Umum dan Perumahan Rakyat, Direktorat Jenderal Bina Marga tahun 2017, terdapat 7 jenis kerusakan pada perkerasan jalan lentur yaitu: pelepasan butir, retak kulit buaya (*alligator cracking*), retak blok (*block cracking*), retak memanjang (*longitudinal cracking*), retak melintang (*transverse cracking*), retak tepi (*edge cracking*), dan alur (*rutting*).

#### 1. Pelepasan Butir (*weathering/reveling*)

Pelepasan butir ditandai dengan lepasnya butir agregat pada permukaan jalan beraspal, pelepasan ini dapat disebabkan oleh rendahnya kandungan aspal, campuran aspal yang kurang baik, kurangnya pemadatan, segregasi atau pengelupasan aspal.



Gambar 2. 1 Pelepasan Butir (Dokumen Pribadi)

#### 2. Retak Kulit Buaya (*alligator cracking*)

Jenis kerusakan ini ditandai dengan adanya retak yang saling berhubungan dan membentuk serangkaian kecil yang terjadi di permukaan jalan beraspal dan berbentuk menyerupai kulit buaya. Jenis kerusakan ini umumnya terjadi karena adanya beban kendaraan berat yang berulang.



Gambar 2. 2 Retak Kulit Buaya (Dokumen Pribadi)

### 3. Retak Blok (*block cracking*)

Jenis kerusakan yang ditandai dengan adanya retak yang saling berhubungan dan akan membentuk permukaan menjadi bentuk kotak-kotak yang berbentuk hampir menyerupai bujur sangkar. Akibat utama keadaan ini yaitu karena adanya penyusutan pada lapisan aspal atau dari karakteristik dan temperatur aspalnya dan tidak diakibatkan oleh beban lalu lintas yang ada.



Gambar 2. 3 Retak Blok (Kementrian PUPR 2017:6)

### 4. Retak Memanjang (*Longitudinal Cracking*)

Jenis kerusakan yang mana ditandai dengan adanya retak paralel yang sejajar dengan sumbu jalan, yang mana kondisi ini dapat terjadi yang disebabkan oleh kurang baiknya pembentukan sambungan memanjang, dapat disebabkan karena terjadinya penyusutan pada lapis beton aspal akibat rendahnya temperatur atau adanya penuaan aspal, siklus temperatur harian atau akibat dari gabungan beberapa faktor tersebut.



Gambar 2. 4 Retak Memanjang (Dokumen Pribadi)

#### 5. Retak Melintang (*Transverse Cracking*)

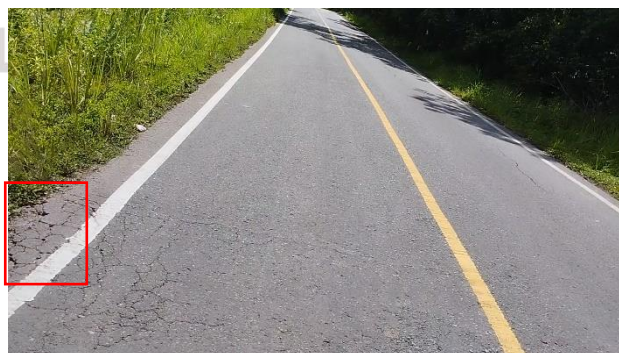
Jenis kerusakan ini ditandai dengan adanya retak dengan arah melebar pada jalan dan arahnya hampir tegak lurus dengan sumbu jalan (arah hamparan jalan). Jenis kerusakan ini biasanya tidak berkaitan dengan tingkat beban lalu lintas yang ada.



Gambar 2. 5 Retak Melintang (Dokumen Pribadi)

#### 6. Retak Tepi (*Edge Cracking*)

Tanda dari jenis kerusakan ini yaitu adanya retak memanjang yang sejajar dengan tepi perkerasan yang mana biasanya terjadi sekitar 0,3 m sampai 0,5 m dari tepi luar perkerasan. Retak tepi dapat diperparah oleh beban kendaraan serta dapat ditimbulkan oleh pelemasan pada lapisan pondasi atas atau pada tanah dasar.



Gambar 2. 6 Retak Tepi (Dokumen Pribadi)

#### 7. Alur (*Rutting*)

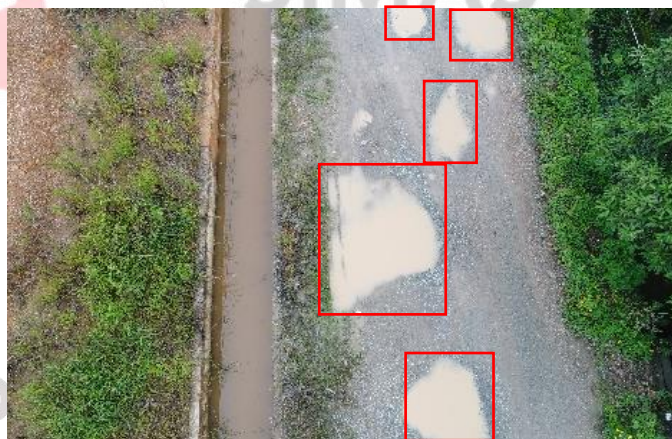
Jenis kerusakan ini, ditandai dengan adanya penurunan memanjang yang terjadi pada Jalur Jejak Roda Kiri (JRKI) dan Jejak Roda Kanan (JRKA) yang mana akibat utama dari kerusakan ini adalah deformasi permanen yang terjadi

pada lapisan permukaan atau tanah dasar yang mana kondisi ini diakibatkan oleh konsolidasi atau pergerakan lateral dari bahan perkerasan akibat dari beban kendaraan yang ada.



Gambar 2. 7 Alur (Kementrian PUPR 2017:6)

Sedangkan jenis kerusakan jalan yang digunakan dalam penelitian ini adalah kerusakan jalan jenis luang (*potholes*), retak struktural (*structural cracking*), retak kulit buaya (*alligator cracking*) dan retak memanjang (*longitudinal cracking*).



Gambar 2. 8 *Potholes* (Dokumen Pribadi)



Gambar 2. 9 *Structural Crack* (Dokumen Pribadi)



Gambar 2. 10 *Alligator Crack* (Dokumen Pribadi)



Gambar 2. 11 *Longitudinal Crack* (Dokumen Pribadi)

## 2.4 Kecerdasan Buatan (*Artificial Intelligence*)

Kecerdasan buatan atau yang biasa disebut dengan *Artificial Intelligence* merupakan suatu kemajuan teknologi yang mengembangkan sistem pemahaman dan berpikir melalui suatu pemrograman yang dimana sistem kecerdasan ini dapat dikatakan sebagai otak tiruan dari manusia yang dapat memecahkan masalah secara kognitif yang umumnya terkait dengan kecerdasan manusia pembuatan AI ditujukan agar dapat mempermudah manusia dalam menyelesaikan suatu pekerjaan (Budiharto, 2018).

Kecerdasan buatan hadir sebagai cabang ilmu dari *Computer Science* yang menjanjikan banyak manfaat dalam menjawab kebutuhan manusia di masa depan. Banyak literatur mengenai *Artificial Intelligence* menyebutkan bahwa ide mengenai kecerdasan buatan diawali pada awal Abad ke-17 ketika *Rene Descartes*

mengemukakan bahwa tubuh hewan bukanlah apa-apa melainkan hanya mesin-mesin yang rumit. Kata “*Intelligence*” sendiri berasal dari bahasa Latin “*Intelligo*” yang berarti “saya paham”. Berarti dasar dari *intelligence* adalah kemampuan untuk memahami dan melakukan aksi. *Intelligence* merupakan istilah yang kompleks yang dapat didefinisikan dengan ungkapan yang berbeda seperti logika, pemahaman, pembelajaran, penyelesaian masalah, dan lain sebagainya. Sedangkan *artificial* sendiri berarti sesuatu yang tidak nyata, seperti tipuan karena merupakan hasil simulasi (Budiharto, 2018).

#### 2.4.1 *Machine Learning*

*Machine learning* merupakan teknik yang digunakan untuk melakukan inferensi terhadap data dengan pendekatan matematis. Inti dari *Machine Learning* adalah untuk memuat model (matematis) yang merefleksikan pola-pola data (Putra, 2020). Dalam *machine learning* terdapat beberapa cabang yang dikategorikan, diantaranya yaitu: *supervised learning*, *semi-supervised learning*, *unsupervised learning* dan *reinforcement learning*.

##### a. *Supervised Learning*

*Supervised learning* adalah pembelajaran sistem yang dilakukan dengan terarah/terawasi. Yang artinya, pada pembelajaran ini terdapat guru yang mengajar (mengarahkan) dan siswa yang diajar. Kita yang membuat sistem berperan sebagai guru, sedangkan mesin berperan sebagai siswa. Dalam *supervised learning* terdapat 2 kategori fungsi yaitu:

##### - Regresi dan Klasifikasi

Pada persoalan regresi, tujuan kita adalah untuk memprediksi nilai output yang bersifat kontinu. Misalnya, dalam regresi fungsi polinomial, kita ingin mencari tahu fungsi  $f(x)$  diberikan data  $\{(x_1, y_1), \dots, (x_N, y_N)\}$ . Setelah itu, kita gunakan fungsi *aproksimasi* untuk mencari tahu nilai  $y_{N+1}$  dari data baru  $x_{N+1}$ . Perbedaan regresi dan klasifikasi adalah pada tipe output.

Untuk regresi, tipe output adalah nilai kontinu, sementara tipe output pada persoalan klasifikasi adalah suatu objek pada himpunan. Tetapi kita dapat mengkonversi fungsi regresi menjadi fungsi klasifikasi. (Putra, 2020 dalam jurnal Maulana Firman, 2021).

b. *Semi-supervised learning*

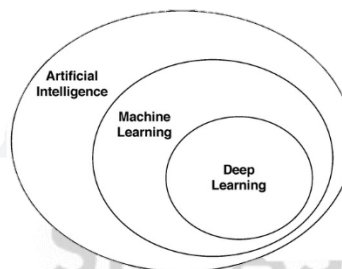
*Semi-supervised learning* memiliki konsep atau metode yang mirip dengan *supervised learning*, yang berbeda pada kedua metode ini adalah pada pelabelan data. Pada *supervised learning*, ada “guru” yang harus membuat kunci jawaban (*input-output*). Sedangkan pada *semi-supervised learning* tidak ada “kunci jawaban” *eksplisit* yang harus dibuat guru. Kunci jawaban ini dapat diperoleh secara otomatis (misalnya dari hasil *clustering*). Pada kategori pembelajaran ini, umumnya kita hanya memiliki sedikit data. Kemudian kita menciptakan data tambahan baik menggunakan *supervised* ataupun *unsupervised learning*, kemudian membuat model belajar dari data tambahan tersebut. (Putra, 2020 dalam jurnal Maulana Firman, 2021).

c. *Unsupervised Learning*

*Supervised learning* melibatkan pengajaran dari seorang guru, maka pada *unsupervised learning* tidak membutuhkan bimbingan dari guru. Sebagai contoh, dalam metode *unsupervised learning* adalah terdapat pada permasalahan *clustering*. Dimana data dikelompokkan tanpa memiliki informasi label sebelumnya. *Clustering* sendiri merupakan salah satu bentuk dari *unsupervised learning*, dan merupakan hasil dari inferensi persamaan. (Putra, 2020 dalam jurnal Maulana Firman, 2021).

### 2.4.2 *Deep Learning*

*Deep learning* adalah suatu cabang khusus dalam bidang *machine learning* yang ditandai dengan penggunaan model yang memiliki banyak lapisan (layer). Pendekatan ini dianggap lebih mendalam karena mengandalkan *neural network*, merupakan model matematis yang terinspirasi dari jaringan syaraf biologis. Yang dimana tujuannya adalah untuk mempelajari representasi data yang ada agar dapat menganalisis data dengan lebih efektif. (Triardhana, 2021).



Gambar 2. 8 Hubungan *AI*, *Machine Learning* dan *Deep Learning*

## 2.5 YOLO (*You Only Look Once*)

Seiring berkembangnya waktu, kemajuan dibidang teknologi pun banyak dikembangkan salah satunya adalah dalam bidang *image processing*. Terdapat banyak peneliti yang telah mengembangkan pada bidang tersebut dengan algoritma dan juga metode yang berbeda-beda. Salah satu algoritma yang banyak digunakan untuk *image processing* atau deteksi dan klasifikasi suatu citra adalah *faster R-CNN*, *Fast R-CNN*, algoritma *SIFT*, dan lain sebagainya, namun untuk dapat menggunakan algoritma tersebut kita memerlukan biaya yang tidak murah karena memiliki kompleksitas dan akurasi yang tinggi serta arsitektur yang kompleks sehingga dapat memperlambat pada proses komputasi atau pemrosesan gambar oleh karena itu diperlukan spesifikasi komputer yang memadai (Aini dkk, 2021).

Redmon dkk (2015) mulai memperkenalkan model algoritma pemrosesan gambar baru bernama 'YOLO' (*you only look once*) dalam papernya. Yolo v1 ini menunjukkan hasil akurasi yang tinggi dan juga cepat dalam mendeteksi, Redmon berusaha menggabungkan proses-proses identifikasi gambar yang ada pada model

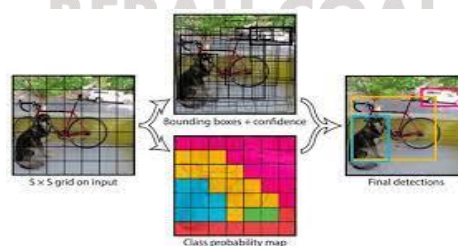
sebelumnya (misal pada *R-CNN*) dilakukan secara terpisah, menjadi sebuah algoritma yang menggunakan *single neural network*, hasil dari penggabungan proses ini ternyata dapat menghasilkan model algoritma yang sangat cepat melebihi model-model yang ada pada saat itu. Namun kecepatan deteksi yang diperoleh pada saat itu tidak sebanding dengan akurasi yang diperoleh yang dimana YOLO v1 hanya mampu memperoleh akurasi sebesar 88% (Aini dkk, 2021).

Kemudian pada tahun 2016, Redmon, dkk akhirnya merilis kembali YOLO v2 dan YOLO9000. Pada dasarnya YOLO v2 memfokuskan pengembangan pada perbaikan *recall* dan lokalisasi bersamaan dengan mempertahankan akurasi klasifikasinya. Selanjutnya pada tahun 2018, Redmon, dkk berhasil meluncurkan kembali versi terbaru dari YOLO yaitu YOLO v3. YOLO v3 menerapkan regresi logistik pada *bounding box* untuk dapat mendeteksi objek lebih baik. Dibandingkan dengan darknet 19 yang digunakan pada YOLO v2 darknet yang digunakan pada YOLO v3 ini lebih baik dengan menggunakan darknet 53 yang memiliki ukuran tertinggi pada *floating point operation* per detiknya yang berarti dapat memanfaatkan GPU dengan lebih baik sehingga lebih efisien dan lebih cepat. Namun, dibalik performa yang lebih tinggi dari versi sebelumnya, YOLO v3 masih masih mengalami beberapa kesulitan, satu diantaranya yaitu kesulitan dalam mengenali objek-objek berukuran medium dan besar (Aini dkk, 2021).

Selanjutnya YOLO v4 dirilis pada tahun 2020 oleh Alexey, dkk. Redmon tidak melanjutkan penelitiannya dalam bidang *computer vision* karena dia beranggapan bahwa hasil karyanya dapat disalahgunakan oleh pihak-pihak tertentu. YOLO v4 lebih menekankan pada pengoptimisasian komputasi paralel dan kecepatan operasi. YOLO v4 ditujukan untuk dapat menggunakan GPU (*Graphical Processing Unit*) konvensional tunggal. Oleh karena itu Alexey, dkk., berhasil memodifikasi YOLO v4 sehingga tidak lagi memerlukan GPU mahal untuk dapat melakukan pemrosesan pengenalan objek lagi. Alexey, dkk., menambahkan beberapa metode yang dimodifikasi untuk melengkapi kemampuan YOLO v4, seperti PAN (*Path Aggregation Network*), CBN (*Cross mini-Batch Normalization*), SAM (*Spatial Attention Module*), dan lain sebagainya (Aini, dkk. 2021).

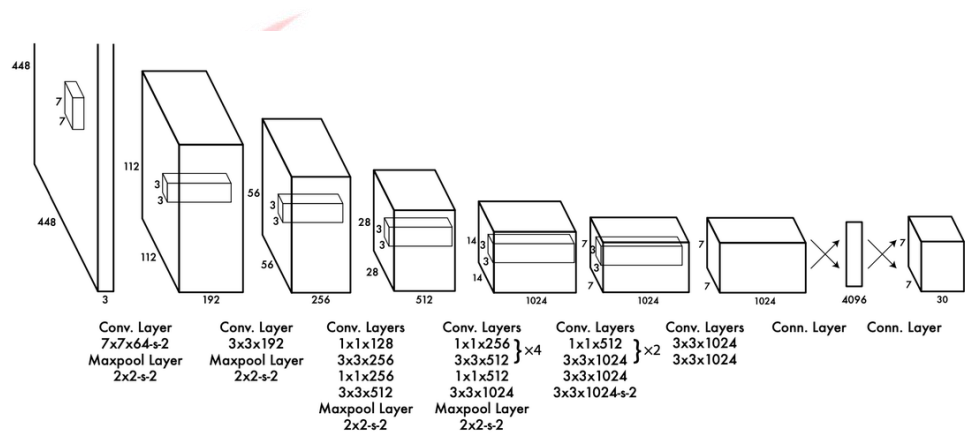
Selanjutnya perkembangan versi dari YOLO ini terus meningkat dengan seiring berjalannya waktu hingga pada saat ini YOLO telah sampai hingga versi 8 dengan tingkat akurasi yang tinggi dan kecepatan deteksi yang lebih baik lagi tentunya. Namun untuk hasil akurasi yang tinggi tentunya memerlukan spesifikasi perangkat yang baik dan mumpuni pula agar hasilnya pun dapat lebih maksimal serta proses komputasi tidak berjalan dengan lambat. Oleh karena itu pada penelitian ini penulis menggunakan YOLO v4 standar yang memiliki akurasi dan kecepatan deteksi yang tinggi dengan *Average precission* meningkat 10% dan FPS meningkat sebesar 12% dari versi sebelumnya, dan YOLO v4 ini tidak menyarankan spesifikasi perangkat yang tinggi dalam pemrosesan citra untuk mendapatkan hasil yang maksimal namun apabila menggunakan spesifikasi perangkat yang mumpuni maka proses *training* atau identifikasi citra dapat berjalan dengan lebih cepat .

YOLO (*You Only Look Once*) merupakan suatu algoritma yang banyak digunakan untuk melakukan deteksi objek secara *real time*. YOLO menggunakan pendekatan yang sangat berbeda dengan algoritma sebelumnya, YOLO menjadikan proses pendekatan objek sebagai masalah regresi tunggal, yang memproses langsung dari piksel gambar hingga koordinat kotak pembatas dan probabilitas kelas. Dengan menggunakan YOLO, sistem hanya melihat sekali (*You Only Look Once*) pada gambar untuk memprediksi benda apa yang ada dan dimana tempatnya. (Redmon, dkk, 2016).

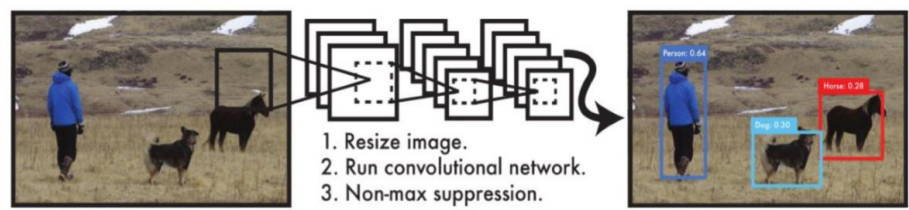


Gambar 2. 9 Model YOLO (Redmon, dkk 2016)

YOLO menyatukan komponen yang terpisah dari pendeteksian objek menjadi satu jaringan saraf. YOLO memanfaatkan fitur dari keseluruhan gambar untuk memprediksi setiap kotak pembatas. YOLO memprediksi semua kotak pembatas pada semua kelas objek untuk sebuah gambar secara bersamaan. Dalam artian YOLO mempertimbangkan seluruh bagian citra secara global dan semua objek pada citra. YOLO membagi gambar masukan menjadi  $S \times S$  petak (*grid*). Jika pusat objek terdapat pada suatu petak, maka sel petak tersebut bertanggung jawab untuk mendeteksi objek tersebut (Pramestya, 2018).



Gambar 2. 10 Arsitektur Dasar YOLO (Redmon, dkk:2016)



Gambar 2. 11 Sistem Deteksi YOLO (Redmon, dkk,2016)

Masing-masing *bounding box* mengandung 5 komponen prediksi yakni  $x$ ,  $y$ ,  $w$ ,  $h$ , dan skor kepercayaan (*confidence*). Koordinat ( $x,y$ ) mewakili titik tengah dari *bounding box* yang telah ditentukan. Sedangkan  $w$  dan  $h$  mewakili nilai lebar (*width*) dan tinggi (*height*) dari sebuah gambar yang kita gunakan. Skor prediksi menyatakan *Intersection Over Union (IOU)* antara kotak yang diprediksi dan *ground truth* (Rofii Faqih. dkk. 2021).

$$\Pr (Class_i | Object) * \Pr (Object) * IOU_{pred}^{Truth} = \Pr (Class_i) * IOU_{pred}^{Truth}$$

Keterangan:

$\Pr (Class_i | Object)$  : probabilitas kondisional kelas  $i$

$\Pr (Object)$  : probabilitas kelas  $i$

Setiap kotak pembatas memiliki skor kepercayaan yang berbeda-beda, ada yang rendah dan ada juga yang memiliki skor keyakinan yang tinggi. YOLO hanya menyimpan skor kepercayaan di atas *threshold* yang nilainya dapat kita tentukan sendiri, misalnya kita memasang *threshold* 20%, maka nilai atau skor keyakinan yang berada dibawah ketentuan *threshold* yang telah kita tentukan, maka prediksi tersebut akan dihilangkan, proses ini disebut dengan *non-max suppression* (Rofii Faqih. dkk. 2021).

## 2.6 Confusion Matrix

*Confusion matrix* dalam *machine learning* disebut sebagai matrik kesalahan. Matrix tersebut berupa tabel yang digunakan untuk mendeskripsikan kemampuan model klasifikasi terhadap data yang tersedia. Jumlah prediksi benar dan salah disimpulkan dengan jumlah nilai setiap kelas. Matriks tersebut dibagi menjadi hasil prediksi model dan kenyataan dilapangan, Nilai TP (*True Positive*) merupakan hasil dari model memprediksi objek secara benar. Nilai FN (*False Negative*) merupakan bentuk objek dalam 1 kelas yang pada kenyataannya positif namun salah dalam memprediksi objek, nilai FP (*False Positive*) merupakan banyak objek dari kelas 2 yang pada kenyataannya termasuk dalam kelas 2 namun model memprediksi sebagai kelas 1. Nilai TN (*True Negative*) merupakan banyaknya objek dalam kelas 2 yang pada kenyataannya termasuk dalam kelas 2 dan model benar dalam memprediksinya (Triardhana, dkk. 2021). Untuk mendapatkan nilai *accuracy*, *precision*, dan *recall* dari *confusion matrix*, maka dapat dilakukan perhitungan dengan rumus sebagai berikut:

$$\begin{aligned}
 \text{Accuracy} &= \frac{TP + TN}{\text{Total dataset}} \\
 \text{Precision} &= \frac{TP}{TP + FP} \\
 \text{Recall} &= \frac{TP}{TP + FN} \\
 \text{F1 Score} &= 2 \times \frac{\text{presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}}
 \end{aligned}$$

keterangan menurut (Lapian, 2021) :

1. *Accuracy*

*Accuracy* merupakan rasio yang diprediksi benar berbanding dengan keseluruhan hasil yang diprediksi oleh sebuah sistem.

2. *Precision*

*Precision* merupakan rasio antara objek yang diprediksi dengan benar berbanding dengan keseluruhan hasil yang diprediksi positif oleh sistem.

3. *Recall*

*Recall* merupakan rasio antara objek yang diprediksi benar berbanding keseluruhan hasil yang sebenarnya.

4. *F1 score*

*F1 score* merupakan perbandingan rata-rata antara precision dan recall yang dibobotkan. *F1 score* sangat berguna dalam evaluasi model klasifikasi, terutama dalam kasus dimana kelas target memiliki distribusi yang tidak seimbang.

|                 |          | True Class |          |
|-----------------|----------|------------|----------|
|                 |          | Positive   | Negative |
| Predicted Class | Positive | TP         | FP       |
|                 | Negative | FN         | TN       |

Gambar 2. 12 Tabel *Confusion Matrix*

## 2.7 Google Colab

Google colab merupakan layanan cloud gratis yang mendukung GPU secara gratis. Google colab bisa dilakukan untuk meningkatkan keterampilan bahasa pemrograman *python*, mengembangkan aplikasi pembelajaran secara mendalam menggunakan perpustakaan yang populer seperti *keras*, *tensorflow*, *pytorch*, dan *openCV*. Fitur google colab yang paling penting dan yang membedakan colab dengan layanan cloud gratis lainnya adalah colab benar benar menyediakan GPU sebesar 12 GB secara gratis (Hidayatulloh, 2021).

## 2.8 Open CV dan Python

*Computer vision* adalah bidang ilmu yang mempelajari cara pandang komputer terhadap objek disekitarnya dengan kemampuan untuk menganalisanya. Hal ini melibatkan penggabungan antara pemrosesan citra digital, pengenalan pola, grafika komputer dan juga pembelajaran mesin. *Computer vision* berada dalam ranah ilmu komputer yang meliputi algoritma, pemrosesan data dan grafik, ilmu fisika seperti optik dan sensor, ilmu matematika termasuk kalkulus dan teori informasi, serta ilmu biologi yang mencakup rangsangan visual dan pemrosesan saraf. *Computer vision* juga merupakan kombinasi antara sistem pencahayaan dan analisis citra atau gambar. Meskipun otak manusia mampu mengenali wajah orang lain bahkan saat pertama kali bertemu, dan mengidentifikasi objek dengan cepat, bagi komputer, sebuah gambar hanya terdiri dari piksel-piksel. Tujuan utama *computer vision* adalah mengenali gambar dengan baik sehingga komputer dapat menghasilkan informasi yang penting dari gambar tersebut. Dengan kata lain, tujuan *computer vision* adalah mengajarkan komputer untuk membuat piksel-piksel menjadi hidup dan terasa seperti di dunia nyata (Manajang, 2020).

*Python* adalah sebuah bahasa pemrograman tingkat tinggi yang bersifat interpreter, interaktif, berorientasi objek, dan dapat digunakan pada hampir semua *platform*, seperti Mac, Linux, dan Windows. *Python* merupakan bahasa pemrograman yang mudah dipelajari karena memiliki sintaks yang jelas, dapat digabungkan dengan modul siap pakai, dan memiliki struktur data tingkat tinggi

yang efisien. Distribusi *Python* dilengkapi dengan fasilitas seperti shell di Linux. Lokasi instalasi *Python* biasanya berada di `"/usr/bin/Python"`, meskipun bisa berbeda-beda. Untuk menjalankan *Python*, kita cukup mengetikkan "*Python*" dan menunggu sejenak hingga muncul tanda `">>>"` yang menandakan *Python* siap menerima perintah. Terdapat juga tanda `"..."` yang menandakan baris berikutnya dalam blok prompt `'>>>'`. Untuk pengembangan *script*, kita dapat menggunakan text editor (Zulkhaidi, 2019).

### 2.8.1 Mengapa bahasa *python*

Perbedaan utama antara *python* dengan bahasa pemrograman lainnya terletak pada aturan penulisan kode program. Hal ini dapat membingungkan para programmer yang tidak terbiasa dengan *python*, terutama dalam hal identasi, tipe data, tuple dan dictionary. *Python* memiliki kelebihan unik dibandingkan dengan bahasa lain, terutama dalam hal penanganan modul, yang membuat beberapa programmer menyukai *python*. Selain itu, *python* juga merupakan produk *open source*, gratis dan dapat digunakan diberbagai platform. (Triasanti, 2000 dalam jurnal Maulana Firman, 2021).

Beberapa fitur yang dimiliki *python* adalah;

- Memiliki keputusan yang luas; dalam distribusi *python* telah tersedia modul-modul siap pakai untuk berbagai keperluan.
- Memiliki tata bahasa yang jernih dan mudah dipelajari. Memiliki aturan *layout* kode sumber yang memudahkan pengecekan, pembacaan kembali dan penulisan ulang kode sumber (berorientasi objek).
- Memiliki sistem pengelolaan memori secara otomatis (*garbage collection*, seperti *java*) modular, mudah untuk dikembangkan dengan menciptakan modul-modul baru; modul-modul tersebut dapat dibangun dengan bahasa *python* maupun *C/C++*.
- Memiliki fasilitas pengumpulan sampah secara otomatis, seperti halnya pada bahasa pemrograman *jaya*.
- Python* memiliki fasilitas pengaturan penggunaan ingatan komputer sehingga para pemrogram tidak perlu melakukan pengaturan ingatan komputer secara langsung.

## 2.9 *Object Detection*

*Object detection* adalah aspek penting dalam kehidupan sehari-hari manusia. Sebagai manusia, kita secara terus menerus melakukan “deteksi” berbagai objek seperti manusia, bangunan, mobil, dan lain sebagainya. Namun, bagaimana kita dapat mendeteksi objek dengan benar dan akurat dengan usaha yang minimal tetap menjadi suatu misteri. Penjelasan yang komprehensif mengenai hal ini telah menjadi perdebatan antara psikolog dan fisiolog selama lebih dari satu abad (Schneiderman, 2000 dalam jurnal Maulana Firman 2021).

*Object detection* memiliki banyak potensi manfaat, termasuk dalam bidang pengambilan gambar. Jumlah koleksi gambar digital telah mengalami peningkatan yang signifikan dalam beberapa tahun terakhir lebih dari 67 juta gambar dalam koleksinya. *The Associated Press* mengumpulkan dan mengarsipkan sekitar 1.000 foto setiap harinya. Namun, saat ini pemanfaatan dari koleksi tersebut terbatas oleh kurangnya metode *retrieval* (pencarian kembali) yang efektif. Untuk menemukan gambar yang spesifik dalam koleksi semacam itu, orang harus mencari menggunakan teks yang berbasis keterangan dan fitur-fitur gambar dasar seperti warna dan tekstur (Schneiderman, 2000 dalam jurnal Maulana Firman, 2021).

*Object detection* dapat digunakan untuk mengekstraksi lebih banyak lagi informasi dari gambar-gambar dan membantu memberi label dan mengkategorikannya. Metode pencarian yang ditingkatkan akan membuat database dapat diakses oleh kelompok pengguna yang lebih luas, seperti lembaga penegak hukum, praktisi medis, desainer grafis, dan multimedia. *Object detection* juga bisa berguna dalam fotografi. Ketika teknologi kamera berubah dari film ke capture digital, kamera akan menjadi bagian optik dan bagian komputer (Schneiderman, 2000 dalam jurnal Maulana Firman, 2021).

## 2.10 *Fotogrametri*

Fotogrametri merupakan salah satu teknik pengukuran dan interpretasi dari citra untuk memperoleh informasi (luas, area, volume dan juga koordinat) dari suatu area atau lokasi terkait objek dengan menggunakan lebih dari satu buah foto. Tujuan utama dari pengukuran fotogrametris adalah untuk mengrekonstruksi objek dalam

tiga dimensi atau untuk keperluan produksi pembuatan peta. Pengolahan data fotogrametris telah mengalami perkembangan pesat dari metode analitik hingga era *computer vision*. Dalam *computer vision*, diperkenalkan metode *Structure from Motion* (SfM) untuk mengolah data foto udara dan menghasilkan model permukaan digital (DSM) dan model topografi digital (DTM), serta *orthophoto*. (Mandaya, 2020).

Berdasarkan definisi tersebut, fotogrametri dapat mencakup dua bidang yaitu fotogrametri metrik dan fotogrametri interpretative (Aries, 2018 dalam jurnal Rifaan Muhammad & Martinus Edwin Tjahjadi, 2019).

a. Fotogrametri Metrik

Fotogrametri metrik atau metrik fotogrametri bertujuan untuk memperoleh data kuantitatif seperti jarak, sudut, luas dan juga posisi dari suatu objek. Untuk memperoleh data tersebut maka diperlukan alat-alat khusus serta pengetahuan dan keterampilan tertentu. Hal ini bertujuan untuk mengetahui hubungan matematis antara sistem foto udara dengan sistem tanah, sehingga ukuran-ukuran yang ada difoto udara dapat dipindahkan ke dalam sistem tanah atau sebaliknya.

b. Fotogrametri Interpretatif

Fotogrametri interpretatif bertujuan untuk memperoleh data kualitatif dengan cara pengenalan, identifikasi dan interpretasi foto udara.

### 2.10.1 *Unmanned Aerial Vehicle* (UAV)

*Unmanned Aerial Vehicle* (UAV) biasa disebut juga sebagai pesawat tanpa awak. UAV fotogrametri merupakan suatu platform untuk pengukuran fotogrametri yang operasinya dapat dikendalikan dari jarak jauh secara semi-otomatis ataupun dapat dilakukan juga secara otomatis tanpa pilot yang berada dalam platform tersebut (Salsabila, 2017 dalam jurnal Rifaan Muhammad & Martinus Edwin Tjahjadi, 2019).

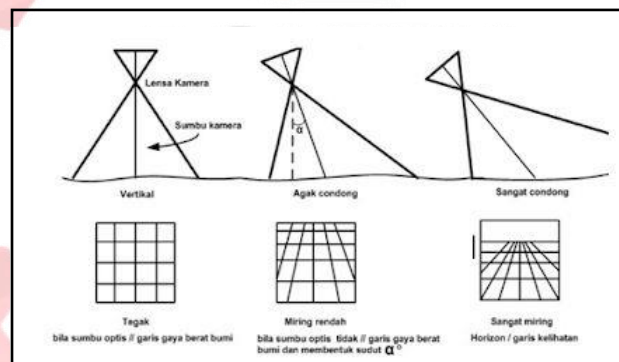
Dalam perkembangannya, UAV telah dimanfaatkan dalam bidang transportasi, salah satunya dalam pengukuran simpang jalan untuk keperluan perencanaan lalu lintas. Pemanfaatan UAV dapat dieksplorasi lebih jauh tidak hanya bidang lalu lintas tetapi juga manajemen sarana transportasi jalan. Berdasarkan hal tersebut, dalam penelitian ini dilakukan eksplorasi kemampuan foto udara yang dipotret menggunakan wahana UAV untuk akuisisi data kerusakan jalan. Jenis drone yang digunakan dalam penelitian ini adalah drone DJI Phantom 4 Advanced dengan ketinggian terbang yang diterapkan adalah tidak lebih dari 15 meter dengan interval pemotretan adalah 3s (detik)/frame (Mandaya, 2020).

Kelebihan dari UAV sendiri dibandingkan dengan pesawat yang dapat dikendarai oleh pilot (berawak) adalah bahwa UAV dapat digunakan pada situasi dengan resiko tinggi tanpa perlu membahayakan nyawa manusia, pada area yang tidak dapat diakses dan terbang pada ketinggian rendah dibawah awan. Selain itu, salah satu faktor kelebihan UAV adalah biaya. Harga perangkat UAV dan biaya operasionalnya jauh lebih murah jika dibandingkan dengan pesawat berawak (Subakti, 2017 dalam jurnal Rifaan Muhammad & Martinus Edwin Tjahjadi, 2019).

### 2.10.2 Foto Udara

Foto udara dapat dikelompokkan menjadi dua kategori, yaitu foto udara *metrik* dan foto udara *non-metrik*. Foto udara *metrik* adalah foto udara yang datanya dihasilkan dari kamera udara. Kamera udara yang digunakan adalah kamera *metrik* dengan fokus yang telah ditentukan sebelumnya. Kamera udara ini berbeda dengan kamera biasa *non-metrik* yang memungkinkan pengaturan fokus sesuai dengan keinginan. (Syauqani dkk. 2017 dalam jurnal Rifaan Muhammad & Martinus Edwin Tjahjadi, 2019).

Berdasarkan jenis posisi pengambilan data, foto udara dapat dibedakan menjadi dua kategori, yaitu foto tegak dan foto miring. Foto udara tegak dihasilkan saat kamera berada dalam posisi tegak lurus terhadap permukaan bumi saat pengambilan gambar. Sementara itu, foto miring dihasilkan saat kamera berada dalam posisi miring saat pengambilan gambar. Untuk keperluan pemetaan, jenis foto udara yang digunakan adalah foto udara tegak. Untuk jenis pengambilan gambar foto udara, dapat dilihat pada gambar 2.17 (Syauqani dkk. 2017 dalam jurnal Rifaan Muhammad & Martinus Edwin Tjahjadi, 2019).



Gambar 2. 13 Jenis foto udara

### 2.10.3 Software Pengolahan Foto Udara

Software pengolahan foto udara merupakan salah satu aplikasi yang digunakan untuk melakukan pemrosesan data foto udara yang dimana hasil akuisisi data digabungkan dengan menggunakan konsep *Structure from Motion* (SfM) berikut ini merupakan beberapa tahapan atau proses pengolahan data foto udara hingga menjadi sebuah data orthophoto menurut (Mandaya, 2020).

#### a. *Align photo*

Tahap ini merupakan proses pengidentifikasian objek yang sama yang terdapat dalam foto stereo atau beberapa foto yang berbeda. Setelah identifikasi, selanjutnya akan dilakukan perhitungan posisi dan orientasi pada setiap foto, dan kemudian dihitung *sparse point cloud*. Hasil dari tahap ini masih berupa DSM

(*Digital Surface Model*) yang belum padat, yang artinya belum memiliki detail yang lengkap.

**b. Build Dense Cloud**

Pada langkah ini, dilakukan proses penggabungan dan pengisian titik-titik dalam *point cloud* yang telah dihasilkan pada tahap sebelumnya. Hasil dari proses ini adalah terbentuknya model *point cloud* yang lebih padat dan mendetail, sehingga saat divisualisasikan, model tersebut akan menyerupai kondisi di lapangan yang sebenarnya.

**c. Build Mesh**

Membangun *mesh* merupakan salah satu langkah untuk membuat model 3D berupa jaringan yang terbentuk dari titik-titik yang saling berdekatan dan dihubungkan satu sama lain, membentuk segitiga atau *tetrahedron*.

**d. Digital Elevasi Model (DEM)**

DEM adalah data digital yang menggambarkan geometri dari bentuk permukaan bumi atau bagiannya yang terdiri dari himpunan titik-titik koordinat hasil sampling dari permukaan dengan algoritma yang mendefinisikan permukaan tersebut menggunakan himpunan koordinat (Tempfli, 1991 dalam jurnal Wardana, Kurniawan Putra Widya Sawitri Subiyanto & Hani'ah 2019). Model DEM ini bisa dibentuk berdasarkan titik-titik koordinat yang kemudian dihubungkan dengan segitiga tidak beraturan yang mempunyai *texture*, sehingga dapat merepresentasikan permukaan objek yang dimodelkan.

**e. Digital Surface Model (DSM)**

DSM merepresentasikan permukaan model dengan berbagai ketinggian dipermukaan bumi yang mencakup *ground*, vegetasi hingga objek buatan manusia.

**f. Digital Terrain Model (DTM)**

DTM merupakan gambaran permukaan berupa digital dari permukaan tanah tanpa ada bangunan di atasnya (Wardana, 2019).

**g. Mosaik Orthophoto**

Mosaik ortofoto atau orthomosaic adalah hasil penggabungan beberapa ortofoto atau foto udara yang diambil dari sudut pandang tegak lurus, ortogonal dan tidak memiliki kesalahan relief. Biasanya, mosaik ortofoto telah dilengkapi dengan informasi koordinat yang menyatakan lokasi sebenarnya di permukaan bumi.

## **2.11 Sistem Informasi Geografis**

Sistem Informasi Geografis (SIG) merupakan sistem yang dirancang untuk dapat bekerja dengan data yang tereferensi secara spasial atau sebuah data yang memiliki koordinat geografi. Sistem informasi geografis adalah bentuk sistem informasi yang menyajikan informasi dalam bentuk grafis dengan menggunakan peta sebagai antar muka. Sistem informasi geografis tersusun atas konsep beberapa lapisan (*layer*) dan relasi. Kemampuan dasar dari sistem informasi geografis adalah dapat mengintegrasikan berbagai operasi basis data seperti *query*, menganalisisnya serta menampilkannya dalam bentuk pemetaan berdasarkan letak geografisnya. (Prahasta, 2009 dalam jurnal Rohim. dkk, 2015).