

BAB II

LANDASAN TEORI

1.1 Tinjauan Pustaka

Dalam penelitian ini akan digunakan tinjauan pustaka yang dapat mendukung penelitian, berikut ini merupakan tinjauan pustaka yang diambil:

1. (Irfan & Widodo, 2020) melakukan penelitian dengan judul “Penerapan Metode Deep Learning Convolution Neural Network pada Robot KRSBI Humanoid R-SCUAD”. Dalam penelitiannya membahas tentang metode *deep learning* untuk melakukan deteksi bola. Menggunakan 6 hidden layer yang diimplementasikan menggunakan framework Tensorflow. Dataset yang digunakan terbagi menjadi 2 *class*, yaitu objek bola dan objek bukan bola. Metode yang digunakan dinilai cukup efektif untuk mengenali bola. Namun masih harus dikembangkan. Dari hasil pengujian, didapat tingkat akurasi deteksi sebesar 67% dengan kecepatan pengolahan gambar maksimal adalah 13 FPS.
2. (Rudiawan, et al., 2017) melakukan penelitian dengan judul “The Deep learning Development for Real-Time Ball and Goal Detection of Barelang-FC”. Dalam penelitiannya membahas tentang deteksi objek real-time yang terintegrasi untuk robot sepak bola humanoid yaitu mendeteksi bola dan gawang dengan memanfaatkan metode You Only Look Once (YOLO) yang digunakan sebagai metode pendeteksian objek pembelajaran yang mendalam. Eksperimen real-time telah dilakukan di LINUX OS dengan menggunakan papan pengontrol NVIDIA JETSON TX1. Hasil percobaan menunjukkan bahwa metode yang diusulkan mampu mendeteksi dan membedakan objek dalam kondisi pencahayaan yang berbeda, dengan

gangguan dari objek lain, juga dari sudut pengambilan gambar yang berbeda.

3. (Javadi, et al., 2017) melakukan penelitian dengan judul “Humanoid Robot Detection using Deep Learning: A Speed-Accuracy Tradeoff”. Dalam penelitiannya membahas tentang perbandingan antara 3 arsitektur pada CNN untuk mendeteksi tubuh robot humanoid, yaitu arsitektur LeNet, arsitektur SqueezeNet, dan arsitektur GoogleNet. Dari pengujian hasil training yang menggunakan 4000 dataset gambar robot dan pengaturan parameter training yang tidak terlalu jauh bedanya. Didapat tingkat akurasi yang sangat tinggi, diantaranya LeNet memiliki rata – rata akurasi sebesar 94.83%, SqueezeNet memiliki rata – rata akurasi sebesar 100%, dan GoogleNet memiliki rata – rata akurasi sebesar 99.9%.
4. (Nurfita, 2018) melakukan penelitian dengan judul “Implementasi *Deep Learning* Berbasis Tensorflow Untuk Pengenalan Sidik Jari”. Penelitian ini bertujuan membuat sebuah sistem untuk mendeteksi antara meja dan kursi motif ukiran Jepara pada suatu gambar dan video. Pada penelitian ini dilakukan pengenalan objek meja dan kursi motif ukiran Jepara menggunakan *framework Tensorflow* dengan *dataset* sebanyak 500 gambar. Hasil penelitian menunjukkan bahwa dengan metode *CNN* didapatkan tingkat akurasi hingga 98% untuk melakukan deteksi meja dan kursi motif ukiran Jepara pada sebuah *frame* gambar dan video
5. (Boschi, et al., 2020) melakukan penelitian dengan judul “A Cost-Effective Person-Following System for Assistive Unmanned Vehicles with Deep Learning at the Edge”. Penelitian ini bertujuan untuk

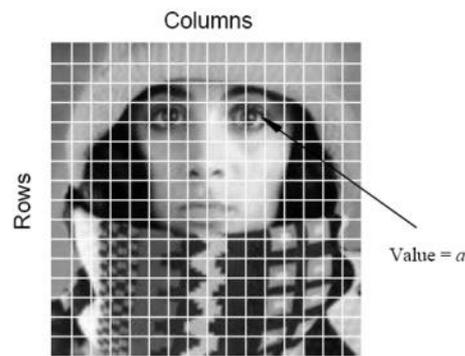
mengetahui perangkat apa yang efektif untuk digunakan di system Person Following pada kendaraan tanpa awak. Penelitian ini membandingkan beberapa perangkat yaitu Raspberry-Pi 3 (RP3), RP3 + Neural Compute Stick (NCS), RP3 + NCS 2, Jetson Nano, Jetson AGX Xavier, RP3 + Coral USB Accelerator. Dengan menggunakan model yang ditraining pada PC yang memiliki spesifikasi VGA RTX 2080ti dan 64gb RAM. Hasil yang paling efektif dimiliki oleh Jetson AGX Xavier dan RP3 + Coral USB Accelerator dengan hasil 30+ fps dan akurasi diatas 70%

1.2 Detection

Deteksi berarti penentuan ada atau tidak adanya suatu obyek pada citra/gambar yang merupakan tahap awal dalam interpretasi citra. Keterangan yang diperoleh pada tahap deteksi bersifat global. Keterangan yang diperoleh pada tahap interpretasi selanjutnya, yaitu pada tahap identifikasi, bersifat setengah rinci. Keterangan rinci diperoleh dari tahap akhir interpretasi, yaitu tahap analisis (Lintz Simonett, 1976).

1.3 Image / Citra

Image atau citra merupakan sebuah data digital berupa gambar yang dihasilkan oleh sensor berupa kamera. Dalam pengolahan data digital, *image* direpresentasikan dalam bentuk matriks yang memiliki jumlah baris dan kolom sesuai dengan resolusinya. Pertemuan antara baris dan kolom disebut sebagai *pixel*. Setiap *pixel* berisi gabungan nilai matrik (RGB, Binary, HSL, HSV atau CMYK) yang mewakili warna dari *image*. Sebagai contoh untuk *image* 8 bit berisi nilai pada range 0-255 yang membentuk warna, misalnya 0 untuk hitam dan 255 untuk putih (Arthaya, et al., 2007).



Gambar 2.1 Representasi *Image* sebagai Matriks

1.4 Deep Learning

Deep Learning merupakan salah satu bidang dari *Machine Learning* yang memanfaatkan jaringan syaraf tiruan untuk implementasi permasalahan dengan dataset yang besar yang memanfaatkan banyak *layer* pengolahan informasi nonlinier untuk melakukan ekstraksi fitur, pengenalan pola, dan klasifikasi.

Dalam *Deep Learning*, metode CNN atau *Convolutional Neural Network* sangatlah bagus dalam menemukan fitur yang baik pada citra ke lapisan berikutnya untuk membentuk hipotesis nonlinier yang dapat meningkatkan kekompleksitasan sebuah model. Model yang kompleks tentunya akan membutuhkan waktu pelatihan yang lama sehingga di dunia *Deep Learning* penggunaan GPU sudah sangatlah umum (Danukusumo, 2017).

1.5 Neural Network

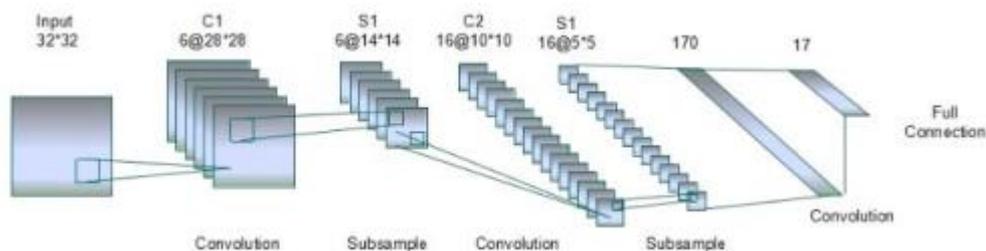
Jaringan saraf tiruan adalah sistem pemrosesan informasi yang memiliki karakteristik kinerja tertentu yang sama dengan jaringan saraf biologis. Jaringan saraf tiruan telah dikembangkan sebagai generalisasi model matematika dari pemahaman manusia atau saraf biologi, berdasarkan asumsi bahwa (Fausett, 1994)

:

1. Pengolahan informasi terjadi pada banyak elemen sederhana yang disebut neuron.
2. Sinyal dilewatkan antara neuron melalui jalur yang terhubung.
3. Setiap jalur yang berhubungan memiliki bobot.
4. Setiap neuron berlaku fungsi aktivasi (biasanya non linier) untuk menentukan sinyal *output*.

1.6 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) merupakan salah satu pengembangan dari jaringan syaraf tiruan yang terinspirasi dari jaringan syaraf manusia yang biasa digunakan untuk mendeteksi dan mengenali suatu objek pada sebuah gambar. CNN terdiri dari neuron yang memiliki bobot, bias dan fungsi aktivasi, (Nurfita, 2018).

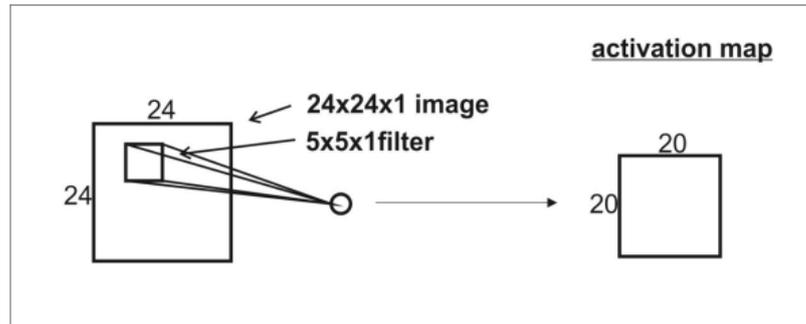


Gambar 2.2 Proses *Convolutional Neural Network*

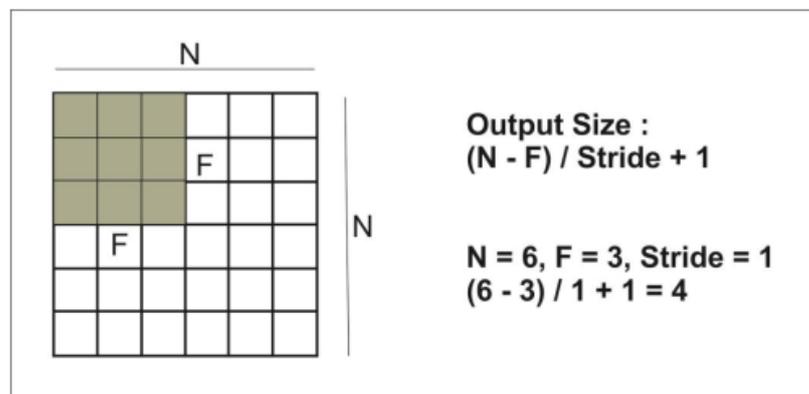
1.6.1 Convolution Layer

Convolutional Layer bagian yang melakukan operasi konvolusi yaitu mengkombinasikan linier *filter* terhadap daerah lokal. *Layer* ini yang pertama kali menerima gambar yang diinputkan pada arsitektur. Bentuk *layer* ini adalah sebuah *filter* dengan panjang (pixel), tinggi (pixel), dan tebal sesuai dengan *channel image* data yang diinputkan. Ketiga *filter* ini akan bergeser keseluruh bagian gambar. Pergeseran tersebut akan melakukan operasi “dot”

antara input dan nilai dari *filter* tersebut sehingga akan menghasilkan output yang disebut sebagai *activation map* atau *feature map*. (Nurfita, 2018).



Gambar 2.3 Proses Konvolusi

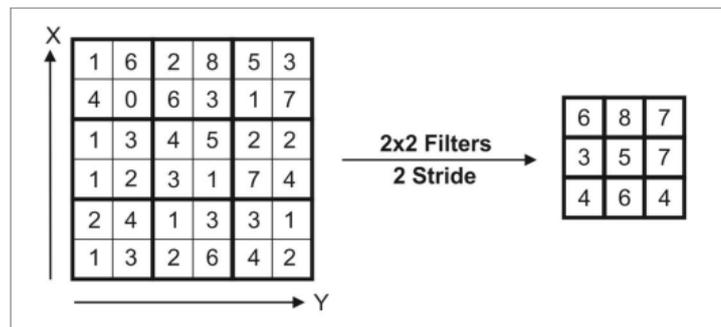


Gambar 2.4 Rumus Menghitung Konvolusi

1.6.2 Pooling Layer

Pooling layer menerima output dari *convolutional layer*, pada *layer* ini ukuran data citra akan direduksi. Prinsipnya *pooling layer* terdiri dari *filter* dengan ukuran tertentu dan *stride/langkaah* kemudian bergeser keseluruh area *feature map*. Sebagian besar arsitektur CNN, metode *pooling* yang digunakan adalah *max pooling*. *Max pooling* membagi output *convolutional layer* menjadi beberapa grid kemudian setiap pergeseran *filter* akan mengambil nilai terbesar dari setiap grid. Tergantung pada panjang langkahnya, gambar yang dihasilkan adalah sebagian kecil dari ukuran aslinya yang berguna untuk

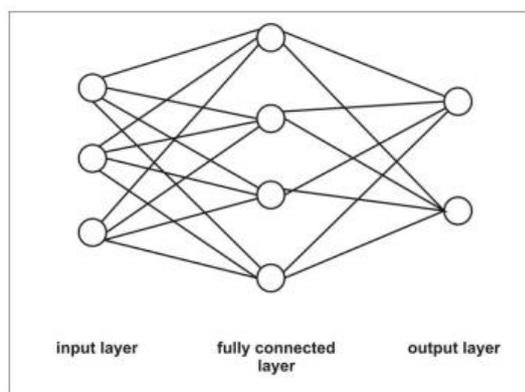
mengurangi dimensi data, sehingga mengurangi jumlah parameter pada langkah selanjutnya (Shukla, & Fricklas, 2018).



Gambar 2.5 Proses Pooling Layer

1.6.3 Fully Connected Layer

Fully connected layer mengambil input dari hasil output *pooling layer* yang berupa *feature map*. *Feature map* tersebut masih berbentuk multidimensional array maka lapisan ini akan melakukan *reshape feature map* dan menghasilkan vektor sebanyak n-dimensi dimana n adalah jumlah kelas output yang harus dipilih program. Misalnya lapisan terdiri dari 500 neuron, maka akan diterapkan softmax yang mengembalikan daftar probabilitas terbesar untuk masing-masing 10 label kelas sebagai klasifikasi akhir dari jaringan (Dutt & Dutt, 2017).



Gambar 2.6 Proses Fully Connected Layer

1.7 TensorFlow

TensorFlow adalah perpustakaan perangkat lunak, yang dikembangkan oleh Tim Google Brain dalam organisasi penelitian Mesin Cerdas Google, untuk tujuan melakukan pembelajaran mesin dan penelitian jaringan syaraf dalam. TensorFlow kemudian menggabungkan aljabar komputasi teknik pengoptimalan kompilasi, mempermudah penghitungan banyak ekspresi matematis dimana masalahnya adalah waktu yang dibutuhkan untuk melakukan perhitungan. Fitur utamanya meliputi (Dewi, 2018):

1. Mendefinisikan, mengoptimalkan, dan menghitung secara efisien ekspresi matematis yang melibatkan array multi dimensi (tensors).
2. Pemrograman pendukung jaringan syaraf dalam dan teknik pembelajaran mesin
3. Penggunaan GPU yang transparan, mengotomatisasi manajemen dan optimalisasi memori yang sama dan data yang digunakan. Tensorflow bisa menulis kode yang sama dan menjalankannya baik di CPU atau GPU. Lebih khusus lagi, TensorFlow akan mengetahui bagian perhitungan mana yang harus dipindahkan ke GPU.
4. Skalabilitas komputasi yang tinggi di seluruh mesin dan kumpulan data yang besar.

1.8 Kontes Robot Sepak Bola Humanoid Indonesia

Kontes Robot Indonesia (KRI) merupakan kegiatan tahunan yang diselenggarakan rutin setiap tahun oleh Kementerian Riset Teknologi dan Pendidikan Tinggi (RISTEK DIKTI). Peserta kompetisi ini berasal dari Perguruan Tinggi Negeri dan Swasta dari seluruh Indonesia. Salah satunya divisi KRSBI Humanoid dimana robot sepak bola ini harus mampu mencari bola, menendang bola dan mencetak goal ke gawang lawan.