BAB 1
PENDAHULUAN

1.1. Latar Belakang Masalah

Pengenalan objek bertujuan menentukan kelompok atau kategori objek berdasarkan ciri-ciri yang dimiliki oleh suatu pola objek. Dengan kata lain, pengenalan objek membedakan suatu objek dengan objek lain. Pendekatan pengenalan objek dalam *computer vision* telah banyak diimplementasikan terhadap berbagai bidang, seperti pengenalan wajah, pengenalan citra karakter atau biasa disebut dengan *optical character recognition* (OCR), bidang kelimuan robotika (seperti robot yang dapat menembak objek tertentu secara otomatis) dan lain sebagainya. Objek yang akan dikenali dapat berupa objek tersebar, seperti objek yang memiliki bentuk dan posisi yang berbeda–beda, selain itu objek tersebar dalam sebuah citra digital memungkinkan memiliki beragam perbedaan translasi (*translation*), mempunyai beragam perbedaan rotasi (*rotation*), serta penskalaan citra (*scaling*).

Neural network backpropagation dapat menjadi solusi permasalahan tersebut, karena *Neural network backpropagation* merupakan salah satu Algoritma *Machine Learning* yang dapat mengenal pola dan mengidentifikasi suatu objek citra berdasarkan pelatihan yang diberikan. *Neural network backpropagation* adalah jaringan dari sekelompok pemproses elemen, unit atau node-node yang dimodelkan berdasarkan neuron otak [1]. Kemampuan pemproses disimpan di dalam kekuatan koneksi antar neuron atau bobot (*weights*), kemampuan tersebut diperoleh dari proses "adaptasi terhadap" atau "belajar dari" suatu set pola pelatihan [1].

Implementasi *neural network backpropagation* memungkinkan mengenal banyak objek citra dalam satu arsitektur *neural network*. Akan tetapi implementasi *neural network backpropagation* ketika digunakan untuk citra yang memiliki beragam translasi, rotasi dan penskalaan citra tanpa dimanipulasi terlebih dahulu, memungkinkan terjadinya proses pembelajaran yang memerlukan jumlah sampel yang cukup banyak.

Algoritma yang memungkinkan menjadi solusi dari beragam translasi, rotasi dan penskalaan objek dalam citra digital adalah *Hu Invariant Moments* [2]. Algoritma ini pertama kali dipublikasikan oleh Hu pada tahun 1961 [2]. *Hu Invariant Moments* merupakan perangkat popular untuk mendeskripsikan ciri geometri pada sistem indentifikasi objek dan pengenalan karakter. Nilai-nilai yang dihasilkan dari *Hu Invariant Moments* ini terdiri dari tujuh nilai yang menidentifikasikan ciri dari sebuah objek citra digital. Nilai-nilai tersebut bersifat independen terhadap translasi, rotasi dan penskalaan. Hal ini dapat dimanfaatkan sebagai masukan dari *Machine Learning* guna mengenali suatu objek.

Berdasarkan uraian pada latar belakang masalah, tujuan dari tugas akhir ini yaitu, melakukan implementasi salah satu *Machine Learning* yaitu *Neural Network Backpropagation* dengan menggabungkan metode *Invarian Moment Hu* sebagai masukan *machine learning* guna mengenali dan mengklasifikasi objek-objek pada citra digital dua dimensi. Penelitian sebagai topik untuk tugas akhir ini mengangkat judul "**IMPLEMENTASI NEURAL NETWORK MENGGUNAKAN HU INVARIANT MOMENT UNTUK MENGENALI OBJEK TERSEBAR**".

1.2. Rumusan Masalah

Berdasarkan latar belakang, identifikasi masalah dalam penulisan penelitian ini adalah bagaimana mengenali objek yang tersebar acak dan memiliki beragam translasi, rotasi dan penskalaan citra dengan data masukan pembelajaran yang seminimal mungkin.

1.3. Maksud dan Tujuan

Maksud dari penulisan tugas akhir ini yaitu, melakukan implementasi salah satu *machine learning* yakni *neural network backpropagation* dengan bantuan *invarian moment hu* yang independen terhadap translasi, rotasi dan penskalaan citra guna mengenali dan mengklasifikasi objek citra tersebar.

Sedangkan tujuan yang ingin dicapai adalah mengetahui hasil akurasi penggabungan *neural network* dan *invarian moment hu* pada pengenalan objek tersebar.

1.4. Batasan Masalah

Batasan masalah yang diterapkan dalam implementasi neural network menggunakan *hu invariant moment* untuk mengenali objek tersebar ini antara lain:

a. Sistem yang dibangun berbasis desktop.
b. Input citra berupa citra tidak bergerak.
c. Objek yang akan dikenali hanya objek yang memiliki ciri empat sudut (persegi / kotak).
d. Objek memiliki warna pinggiran yang kontras dari latar belakangnya.
e. Objek dalam citra tidak bertumpuk dan tidak terpotong.
f. Dalam implementasi sistem harus menggunakan kamera (*webcam*) miniman 3 Mega Pixel (disarankan 5 Mega Pixel atau di atasnya).

1.5. Metodologi Penelitian

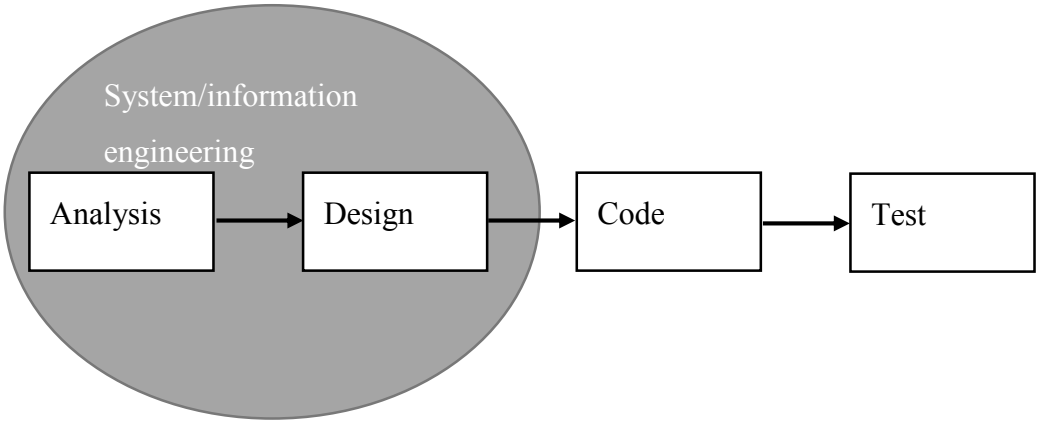
Metodologi penelitian yang digunakan dalam implementasi *Neural Network Menggunakan Hu Invariant Moment Untuk Mengenali Objek Tersebar* ini adalah metode deskriptif. Metode deskriptif merupakan metode yang menggambarkan fakta-fakta dan informasi dalam situasi atau kejadian dimana sekarang secara sistematis, faktual dan akurat. Metode penelitian ini memiliki dua tahapan, yaitu tahap pengumpulan data dan tahap pengumpulan perangkat lunak. meliputi beberapa proses diantaranya :

1.5.1. Metode Pengumpulan Data

Metode pengumpulan data yang digunakan dalam penelitian ini adalah studi literatur, yaitu pengumpulan data yang bersumber dari buku, jurnal, paper dan situs internet untuk dijadikan referensi penulisan dan penelitian.

1.5.2. Metode Pembangunan Perangkat Lunak

Dalam membangun sistem ini, digunakan metode pengembangan perangkat lunak dengan menggunakan metode waterfall sebagai berikut :



Gambar 1.1 Waterfall model [9]

a. *System / Information engineering and modeling.*

Permodelan ini diawali dengan mencari kebutuhan dari keseluruhan sistem yang akan dianalisis ke dalam bentuk *software*. Kebutuhan tersebut terdiri dari pemilihan bahasa pemprograman, pemilihan *framework*, materi-materi algoritma dasar *Neural Network*, *Hu Invarian Moment,* penghalusan citra dan pendeteksian kotak (deteksi tepi Canny dan deteksi *corner* Harris), selain itu mencari kebutuhan yang akan digunakan sebagai bahan uji analisis yakni pengambilan citra objek-objek tersebar. Hal ini sangat penting, mengingat *software* harus dapat dimodelkan, diimplementasikan dan diuji untuk mengetahui akurasi algoritma yang dianalisis.

b. *Software Requirements Analysis*.

Proses pencarian kebutuhan diintensifkan dan difokuskan pada *software*. Untuk mengetahui sifat dari program yang akan dibuat, maka diperlukan pemahaman tentang domain informasi dari *software*, misalnya fungsi yang dibutuhkan, metode yang dibutuhkan, *user interface*, dsb. Proses ini mengacu pada aktivitas sebelumnya (*System / Information engineering and modeling)* kemudian didokumentasikan.

c. *Design*

Proses ini digunakan untuk mengubah kebutuhan-kebutuhan berdasarkan kedua proses sebelumnya menjadi representasi ke dalam bentuk "*blueprint*" software sebelum *coding* dimulai. Proses desain membantu mengimplementasikan kebutuhan-kebutuhan yang telah disebutkan pada tahap sebelumnya. Seperti aktivitas sebelumnya, maka proses ini juga didokumentasikan sebagai konfigurasi dari software.

d. *Code Generation*

Proses ini bertujuan untuk mengubah hasil design hingga dapat dimengerti oleh mesin. Mesin dalam hal ini adalah komputer. Bentuk yang dimaksud yakni ke dalam bentuk bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap *design* secara teknis.

e. Testing.

Semua fungsi-fungsi software harus diujicobakan, agar *software* bebas dari *error*, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya sehingga implementasi algoritma dalam software dapat dianalisis.

1.6. Sistematika Penulisan

Sistematika penulisan tugas akhir ini disusun untuk memberikan gambaran secara umum tentang penelitian yang dijalankan. Sistematika penulisan tugas akhir ini adalah sebagai berikut :

BAB 1 PENDAHULUAN

Bab ini akan membahas mengenai latar belakang masalah, perumusan masalah, maksud dan tujuan, batasan masalah, metodologi penelitian yang digunakan, serta sistematika penulisan.

BAB 2 LANDASAN TEORI

Bab ini berisikan berbagai konsep dasar dan teori-teori yang berkaitan dengan topik penelitian yang dilakukan dan hal-hal yang berguna dalam proses analisis permasalahan. Teori-teori yang berkaitan dimulai dari konsep dasar cintra digital, pengolahan citra, pengenalan pola, computer vision dan model-model warna. Berlanjut pada konsep yang dipilih sebagai metode preprosessing citra diantaranya *convolution*, *gaussian*, penghalusan citra (*smoothing)* menggunakan *kernel gaussian*, algoritma *canny* untuk deteksi tepi dan algoritma *harris* untuk deteksi *corner.* Berlanjut pada konsep yang dipilih sebagai metode pengenalan objek dalam citra yakni *Neural Network Backpropagation* dan *Invarian Moment Hu.* Pembahasan teknologi yang digunakan yakni *Compiler GCC* dengan *MingW* juga pemakaian framework QT untuk pembuatan UI, *framework* OpenCV yang digunakan sebagai pengambilan data citra dan QT Creator yang digunakan sebagai UI *editor* dan *code editor* C++. Terakhir pembahasan tools untuk memodelkan sistem yakni UML.

BAB 3 ANALISIS DAN KEBUTUHAN ALGORITMA

Bab ini berisi analisis masalah serta menjelaskan keputusan pemecahan persoalan yang ada. Didalamnya berisi analisis masalah, analisis masalah sistem, analisis data masukan, analisis metode algoritma, analisis kebutuhan perangkat lunak baik itu kebutuhan non fungsional maupun fungsional. Pembahasan pemodelan sistem menggunakan UML dengan mamakai use case, activity diagram, sequence diagram, class diagram.

BAB 4 IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini berisi hasil implementasi dari hasil analisis dan perancangan yang telah dibuat disertai juga dengan hasil pengujian dalam hal ini keakurasian dari implementasi aplikasi.

BAB 5 KESIMPULAN DAN SARAN

Bab ini akan dijelaskan mengenai kesimpulan terhadap hasil penelitian berikut saran-saran