

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Manusia disebut makhluk sosial, karena akan selalu berinteraksi dengan manusia yang lain untuk mengetahui informasi pada lingkungan sekitarnya serta untuk mengetahui apa saja yang terjadi dalam orang lain maupun dirinya sendiri, rasa itulah yang membuatnya memaksa perlunya seseorang untuk berkomunikasi. Manusia sebagai makhluk sosial yang hidup berkelompok, interaksi dan komunikasi tidak dapat dihindari [1]. Diperlukannya komunikasi untuk interaksi antara sesama manusia yang dapat saling dipahami oleh penerima maupun pemberi suatu informasi. Komunikasi merupakan kegiatan yang sering dilakukan antar manusia dengan manusia lain. Komunikasi ialah cara untuk menyampaikan informasi antar individu, kelompok, organisasi, atau masyarakat. Fungsi utama komunikasi ialah untuk menyampaikan informasi, berbagi pendapat, berinteraksi dengan orang lain, mempertahankan hubungan yang baik, dan mengekspresikan diri [2]. Namun tidak semua dapat berkomunikasi dengan baik. Hal ini karena sebagian dari kita mengalami kesulitan dalam berkomunikasi. Baik itu karena penyakit atau memang kurang sempurnanya indra mereka. Meskipun demikian ada cara untuk saling berkomunikasi dengan mereka menggunakan bahasa khusus yang digunakan oleh para tunarungu untuk berkomunikasi yaitu bahasa isyarat. Tetapi tidak semuanya memahami bahasa isyarat. Maka dari itu diperlukan sistem deteksi khusus untuk memudahkan komunikasi antara orang yang memahami bahasa isyarat dengan orang yang kurang memahami bahasa isyarat. Sudah ada alternatif yang telah dikemukakan untuk mengatasi permasalahan ini. Di antaranya ialah sistem deteksi tersebut sudah dikemukakan menggunakan pemrograman *python*.

Sistem deteksi bahasa isyarat menggunakan *python* sudah banyak ditemukan dengan menggunakan pengolah citra digital. Pengolah citra digital memainkan peran penting dalam *mediapipe* karena pengolah citra digital bertanggung jawab atas berbagai tugas pemrosesan citra dan video, dan *mediapipe* digunakan untuk menerapkan berbagai solusi di berbagai aplikasi. Beberapa peran

penting pengolah citra digital dalam mediapipe ialah deteksi objek, pemrosesan citra *real-time*, klasifikasi objek, pengenalan gerakan dengan cepat dan efisien.

Terdapat beberapa penelitian terdahulu yang telah mencoba mengembangkan system deteksi Bahasa isyarat menggunakan *python*. Di antaranya dengan pembahasan sistem deteksi simbol SIBI menggunakan *mediapipe* dan *ResNet-50* penelitian ini membahas tentang mendeteksi alfabet Bahasa isyarat dengan metode yang digunakan CNN (*Convolutional Neural Network*) dengan model *ResNet-50*, kemudian pada percobaan selanjutnya dihitung performa setelah menambahkan teknologi mediapipe pada sistem deteksi. Hasil dari perbandingan tersebut terbukti bahwa menambahkan teknologi *mediapipe* dapat mendapatkan tingkat akurasi hingga data *training* 88% dan akurasi *validation* 87%[3]. Pada penelitian selanjutnya dengan judul *Hand Gesture Recognition* sebagai Penerjemah Bahasa Isyarat yang digunakan pada *Android* penelitian ini membahas tentang sistem deteksi yang berisi SIBI dan BISINDO. Menggunakan sistem deteksi pada mediapipe dapat dioperasikan berbasis android[4]. Kemudian pada penelitian selanjutnya dengan topik Bahasa Isyarat Indonesia (BISINDO) sebagai pengenalan menggunakan Arsitektur *Mobile-net* penelitian ini membahas tentang penerapan metode *Deep Learning* secara optimal menggunakan model *Mobile-Net* guna melakukan pengenalan gambar pada BISINDO beserta menghitung nilai tingkat akurasinya. Berdasarkan pada penelitian-penelitian yang sudah diamati, maka terdapat beberapa kesimpulan yang dapat diambil. Metode *Deep Learning* cukup bagus dengan arsitektur *MobileNetV2* yang terbilang ringan komputasinya mendapatkan hasil akurasi sebesar 94% dan hasil skor presisi, *recall*, serta f1 yang nilainya sama yaitu 95% [5]. Pada penelitian selanjutnya dengan pembahasan *Leap Motion* dengan sensor gerak digunakan untuk membantu bahasa isyarat Tuna Rungu/Wicara menjelaskan tentang penggunaan metode *Leap Motion* pada penerjemah bahasa isyarat yang cukup baik hingga mendapatkan nilai akurasi rata-rata di angka 78% [6]. Terakhir penelitian dengan judul Klasifikasi Teknik Bulutangkis dengan berbagai pose menggunakan CNN (*Convolutional Neural Network*) berisi perbandingan klasifikasi dengan menggunakan berbagai metode yaitu metode *supervised learning* antara lain *logistic regression*, *random forest*, dan

*Convolutional Neural Network* menghasilkan *accuracy* yang sangat baik pada nilai 80% sampai 90% [7].

Berdasarkan permasalahan diatas, maka penelitian ini diusulkan menggunakan metode *Random Forest*. *Random forest* ialah algoritma pemisahan biner rekursif menggunakan salah satu metode *Decision Tree* pada *Machine Learning*. Keunggulan algoritma ini termasuk performa yang baik, tingkat error yang relatif rendah, dan kemampuan untuk menangani data yang sangat besar.

Selain itu, penulis mengimplementasikan model *Mediapipe* oleh *Google*. *Mediapipe* mampu mendeteksi gerakan tangan, tubuh, dan ekspresi pada wajah yang sangat kompleks dengan data komputasi yang efisien. *Training* dan *testing* dengan berbagai jenis bahasa isyarat sehingga membuktikan *mediapipe* dapat digunakan untuk berbagai bahasa isyarat yang lain.

Dari hasil studi yang telah diuraikan di atas, terlihat beragam pemrograman sistem deteksi menggunakan *python*. Namun, sistem deteksi tersebut kebanyakan masih belum digunakan untuk mendeteksi kata secara *real-time*. Percobaan ini menggunakan metode *Random Forest* diharapkan memiliki tingkat akurasi yang tinggi dibandingkan dengan tingkat akurasi yang lebih tinggi dibandingkan metode yang lain. Dugaan sementara hasil dari penelitian ini merupakan sistem deteksi pemrograman *python* yang lebih baik dibandingkan dengan penemuan terdahulunya.

## 1.2 Rumusan Masalah

Dari latar belakang di atas, maka permasalahan penelitian dirumuskan sebagai berikut:

1. Bagaimana cara merancang sistem deteksi penerjemah kata dalam bahasa isyarat secara *real-time* menggunakan metode *RF (Random Forest)*?
2. Bagaimana menguji hasil tingkat akurasi yang diperoleh dari metode *Random Forest* pada sistem deteksi untuk menerjemahkan kata dalam bahasa isyarat?

### **1.3 Tujuan Penelitian**

Tujuan pada penelitian ini untuk menciptakan sistem deteksi penerjemah Bahasa Isyarat berbasis *python* yang dapat mengklasifikasi dalam bentuk kata secara *Real-Time*.

### **1.4 Batasan Penelitian**

Batasan penelitian ini pada pembuatan program sistem deteksi *sign language* penerjemah Bahasa Isyarat dengan menggunakan klasifikasi 10 kata menggunakan *python*.

### **1.5 Manfaat Penelitian**

Manfaat pada penelitian ini untuk membuktikan bahwa sistem deteksi Bahasa isyarat tidak hanya dapat mendeteksi gerakan tangan melainkan juga dapat mendeteksi bagian tubuh yang lain sehingga dapat menghasilkan klasifikasi yang lebih akurat.

### **1.6 Sistematika Penulisan**

Supaya dapat dipahami dengan baik penjelasan tentang penelitian ini. Berikut sistematika yang digunakan penulisan tugas akhir:

#### **BAB I Pendahuluan**

Pada bab pertama ini penulis membahas latar belakang penelitian yang terdiri dari rumusan masalah, tujuan studi terdahulu terkait penelitian yang akan diteliti, hipotesa penelitian, batasan penelitian dan manfaat penelitian.

#### **BAB II Tinjauan Pustaka**

Menjelaskan teori-teori pemahaman pemrograman dan pemilihan Bahasa Isyarat Indonesia yang dapat digunakan pada penelitian.

### **BAB III Metode Penelitian**

Menguraikan tentang *flowchart* kerja penelitian dan metode penelitian yang akan dipakai serta tahapan penelitian serta menjelaskan tentang pemrograman dan spesifikasi perangkat yang digunakan untuk pembuatan sistem.

### **BAB IV Hasil dan Pembahasan**

Membahas hasil dari pengujian sebuah sistem deteksi gerakan manusia untuk mengetahui hasil evaluasi berupa nilai tingkat akurasi dan uji coba secara *real-time* menggunakan metode *Random Forest*.

### **BAB V Penutup**

Membuat kesimpulan tentang hasil dan saran yang didapat terhadap pengembangan yang dapat dilakukan oleh peneliti yang akan menggunakan topik ini di masa mendatang.

