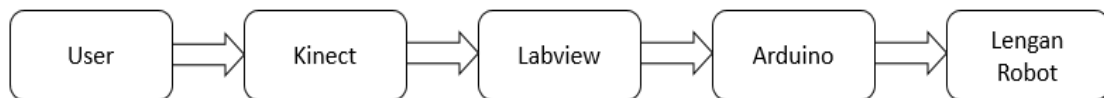


## BAB 3

### METODOLOGI

#### 3.1 Perancangan Sistem

Sistem akan terbagi menjadi dua bagian, proses identifikasi kerangka tubuh menggunakan Kinect pada obyek yang ada dihadapannya. Hasil dari identifikasi dijadikan sebagai masukan pada lengan robot. Proses kedua adalah pengendalian lengan robot agar sesuai yang diperintahkan.



Gambar 3.1 Diagram Alir

##### 3.1.1 Proses Identifikasi Kerangka Tubuh

Kinect memiliki dua kamera dan satu IR *emitter*. Satu kamera mengambil warna sekitar dan yang lainnya mengkalkulasi kedalaman citra (*Depth Image*). IR *emitter* mengirim laser kepada obyek yang ada dihadapannya, dimana laser tersebut akan terpantul kembali ke kamera IR. Kinect mengambil 30 *frame* per detik dengan resolusi 640x480 dan dapat menjalankan dua fungsi kamera warna dan *depth image* secara bersamaan. Obyek yang akan diidentifikasi oleh Kinect harus berada pada jarak dan ketinggian yang tetap agar Kinect dapat memproses warna RGB dan juga *depth image* secara terus-menerus. Kerangka tubuh dapat diidentifikasi dengan mudah akan tetapi tingkat presisinya masih rendah karena memproses beberapa *joint* secara bersamaan menggunakan perangkat lunak yang telah disediakan oleh *Microsoft SDK*. Untuk mendapatkan kerangka tubuh manusia dibutuhkan LabVIEW sebagai GUI (*Graphical User Interface*). Program yang akan digunakan adalah *Kinesthesia Toolkit* yang dikembangkan oleh *University of Leeds*. *Toolkit* tersebut dapat membantu untuk menginsialisasi, mengkonfigurasi dan memproses data-data yang lain.

##### 3.1.2 Proses Pengendalian Manipulator Robot Wayang

*Skeletal tracking* akan dijadikan masukan untuk Robot Wayang. Data masukan dari Kinect kemudian diproses oleh komputer. Keluaran data yang telah diproses pada komputer kemudian akan dikirim ke mikrokontroler Arduino. Saat data telah diterima Arduino, maka kemudian aktuator yang berupa manipulator Robot Wayang akan meniru pergerakan.

## **3.2 Perancangan *Hardware***

### **3.2.1 Perancangan Arduino**

Perancangan Arduino berfungsi sebagai penggerak lengan robot. Komunikasi serial antara komputer dan Arduino terhubung melalui USB. Keluaran Arduino terhubung dengan 4 *motor servo* pada lengan robot.

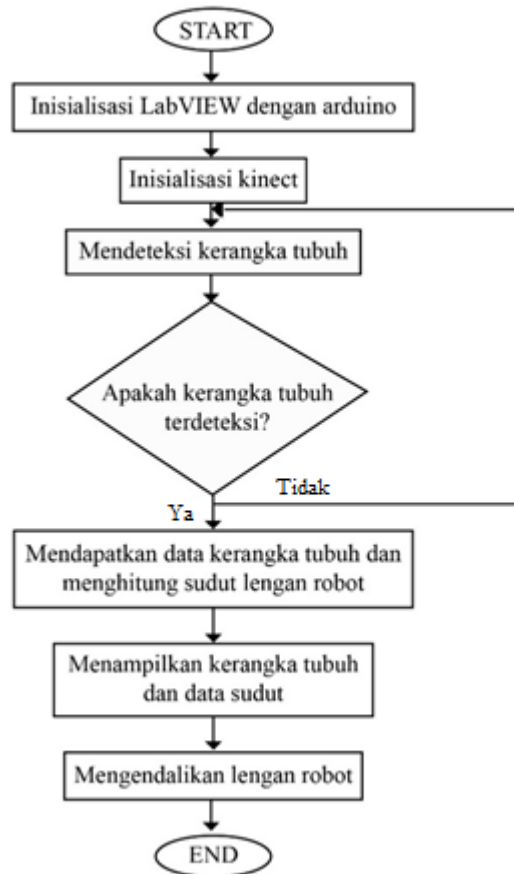
Pin yang digunakan untuk menghubungkan Arduino dengan lengan robot antara lain 0,1,2,3. *Motor servo* yang terhubung pada pin dapat berputar 180°. Pin 0 terhubung dengan *joint base* yang merubah posisi seluruh robot. Pin 2 terhubung dengan *joint* bahu. Pin 1 terhubung dengan *joint* siku. Pin 3 terhubung dengan *joint* pergelangan tangan.

## **3.3 Perancangan *Software***

*Software* yang akan digunakan pada penelitian ini adalah LabVIEW sebagai bahasa pemrograman dan GUI. Pemrograman Arduino hanya digunakan sebagai *driver* agar dapat berinteraksi dengan LabVIEW.

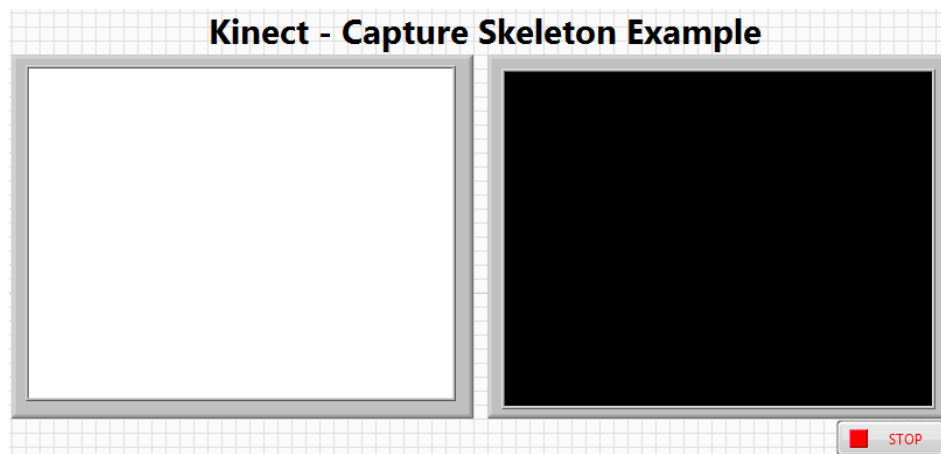
### **3.3.1 Pemrograman LabVIEW**

*Software* LabVIEW digunakan untuk mengelola data kerangka tubuh dari Kinect. *Driver* Kinect dan Arduino perlu dipasang terlebih dahulu agar Kinect dan Arduino dapat dikenali. *Driver* Kinect berupa SDK (*Software Development Kit*). Program Kinect menggunakan program yang dikembangkan oleh *University of Leeds* berupa *toolkit* yang membantu untuk mendapatkan kerangka tubuh disebut dengan *Kinesthesia Toolkit*. Setelah Kinect dapat beroperasi, maka proses selanjutnya dapat dilakukan. *Flowchart* pemrograman LabVIEW dapat dilihat pada gambar 3.2.



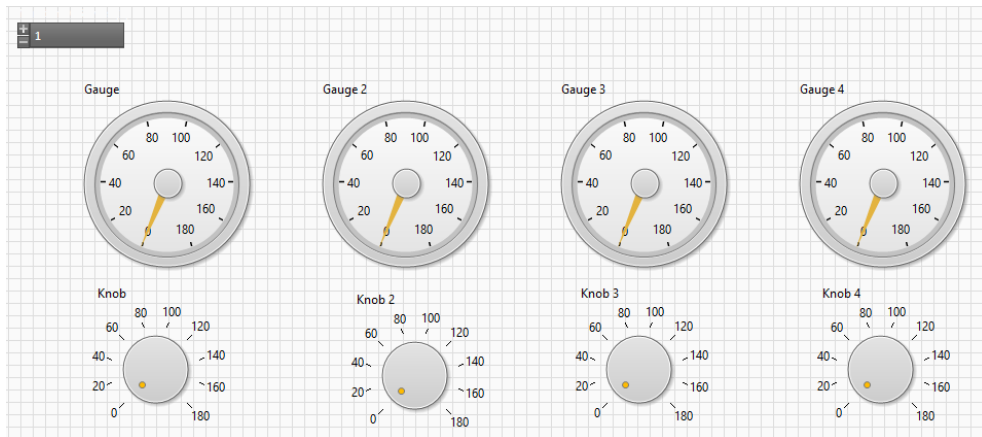
Gambar 3.2 Flowchart pemrograman LabVIEW

Gambar 3.3 menunjukkan sebagian dari *front panel* yang menampilkan video Kinect dan 3D kerangka tubuh. Dengan demikian data kerangka tubuh bisa diperoleh untuk dikelola. Kamera dan kerangka akan terus aktif ketika Kinect berhasil mendapatkan kerangka tubuh dan akan berhenti ketika tombol stop ditekan. *Front panel* memiliki beberapa komponen, yaitu menampilkan video dan kerangka tubuh, pengendalian manual, data keluaran Kinect dari setiap *joint* dan sudut keluaran Kinect dan Arduino.



Gambar 3.3 *Front panel* Kinect

Pengendalian manual menggunakan *gauge* dan *knob* untuk mengubah sudut lengan robot secara manual. *Gauge* berfungsi sebagai keluaran yang menampilkan sudut lengan robot. *Knob* berfungsi sebagai masukan yang menentukan sudut lengan robot. Tetapi sebelum dikendalikan secara manual, jumlah servo motor harus ditentukan terlebih dahulu.



Gambar 3.4 *Front panel* pengendalian manual

Kinect mendapatkan koordinat menggunakan sensor *infrared* untuk menemukan titik *joint* dalam bentuk 3D. Ketika  $x = 0, y = 0, z = 0$  dapat didefinisikan bahwa lokasi *joint* berada di tengah sensor IR. Pada sistem koordinat Kinect,  $x$  mendefinisikan posisi *joint* berada pada kanan sensor.  $y$  mendefinisikan posisi *joint* berada diatas atau dibawah sensor.  $z$  mendefinisikan jarak *joint* dengan jarak sensor. Parameter  $x, y$  dapat bernilai *positive* atau *negative*, tergantung posisinya pada sensor. Tetapi parameter  $z$  akan selalu bernilai *positive*. Koordinat diukur dalam meter. Sensor Kinect dapat merekam dengan jarak maksimal 8 meter, tetapi untuk mendapatkan kerangka tubuh maka jarak maksimalnya diantara 1.5 sampai 4.5 meter. agar Kinect dapat mendeteksi keberadaan manusia, pengguna perlu berdiri menghadap Kinect dengan jarak minimal 1.5m dari Kinect. Jika tidak terdeteksi, maka pengguna perlu mencoba dengan jarak yang lebih jauh. Ketika kerangka tubuh terdeteksi, maka akan tertampil pada *front panel* pada gambar 3.3 dan proses menghitung sudut pada seluruh *joint* terjadi. Perhitung sudut *joint* menggunakan persamaan :

$$\cos \theta = \frac{\vec{ab} \cdot \vec{ac}}{|\vec{ab}| \cdot |\vec{ac}|} \quad ; 0 \leq \theta \leq 180^\circ \quad (3.1)$$

Persamaan 3.1 dapat digunakan untuk menghitung sudut dalam ruang 2D maupun 3D. Pada penelitian ini, persamaan 3.1 digunakan untuk menghitung sudut dalam ruang 3D. Setiap titik

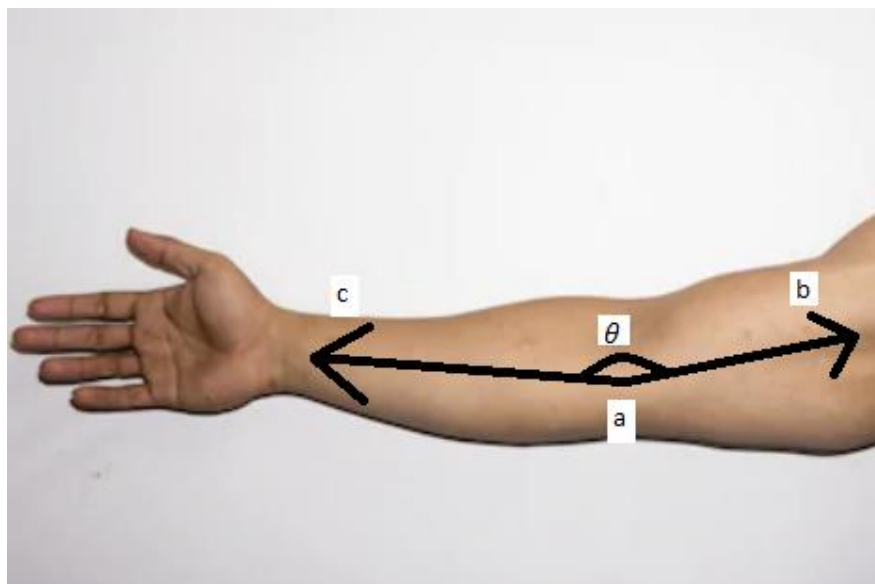
memiliki tiga koordinat yaitu titik  $a(x_1, y_1, z_1)$  dan titik  $b(x_2, y_2, z_2)$  dan titik  $c(x_3, y_3, z_3)$ . Vektor  $\vec{ab}$  dapat dinyatakan dalam bentuk :

$$\vec{ab} = \begin{pmatrix} b_1 - a_1 \\ b_2 - a_2 \\ b_3 - a_3 \end{pmatrix} \quad (3.2)$$

dan vektor  $\vec{ac}$  dapat dinyatakan dalam bentuk :

$$\vec{ac} = \begin{pmatrix} c_1 - a_1 \\ c_2 - a_2 \\ c_3 - a_3 \end{pmatrix} \quad (3.3)$$

Seluruh data koordinat *joint* yang digunakan memiliki nilai  $x, y, z$ . Persamaan 3.1 digunakan untuk mendapatkan sudut *joint* pada manipulator robot. Maka sebagai contoh untuk mengimplementasi persamaan 3.1, *joint* siku sebagai koordinat titik a, *joint* bahu sebagai koordinat titik b dan pergelangan tangan sebagai koordinat c seperti gambar 3.6.



Gambar 3.5 Lengan kanan.

*Joint* yang dihitung besarnya sudut adalah *joint* pada bahu, siku dan pergelangan tangan. *Joint* pada *base* tidak jauh berbeda cara perhitungannya. *Joint* bahu, siku, pergelangan tangan, lengan bergerak secara vertikal, sedangkan *joint* pada *base* lengan bergerak secara horizontal. Orientasi manipulator lengan Robot Wayang pada setiap *joint* bernilai  $180^\circ$ . Setelah mendapatkan seluruh nilai *joint*, nilai tersebut dikirim menuju Arduino melalui komunikasi serial.