

SISTEM PENGHITUNG JUMLAH REPETISI GERAKAN DUMBBELL BICEPS CURLL MENGGUNAKAN MEDIAPIPE DAN OPENCV

Mahesa Tirta Panjalu (Mahasiswa)

Muhammad Achsan Isa Al Anshori, S.Kom., MMSI.

Gramapuri Tamansari Blok C28 No.3 RT.008/RW.033 Wanasari, Cibitung, Bekasi

Mahesat8@gmail.com

Jurusan Informatika, Fakultas Teknologi Teknologi Industri, Universitas Gunadarma

ABSTRAK

Penelitian ini bertujuan untuk mengembangkan sebuah sistem penghitung jumlah repetisi gerakan *dumbbell* menggunakan OpenCV dan Mediapipe. Gerakan repetisi dalam latihan *dumbbell* merupakan komponen penting dalam program latihan kebugaran dan pengembangan kekuatan otot. Namun, menghitung secara manual jumlah repetisi yang dilakukan oleh seseorang memerlukan pemantauan yang konstan dan akurat. Penggunaan *framework* mediapipe adalah kunci dari penelitian ini, sebuah pustaka yang baru saja dikembangkan oleh google untuk mendeteksi sebuah object. Proses deteksi dimulai dengan mendeteksi pose dari gerakan yang tampil pada setiap *frame* video, kemudian menghitung jumlah repetisi berdasarkan perubahan derajat pada bagian siku. Tahap pembuatan sistem ini dimulai dari perencanaan, analisis sistem, perancangan, implementasi dan uji coba sistem. Dimana dilakukan pendeteksian dengan membuat *landmark* pada setiap bagian tubuh yang terlihat pada *frame* video. Selama proses berjalan, terdapat dua kotak dibagian atas yang berfungsi untuk mengitung jumlah repetisi gerakan *dumbbell biceps curl* dan menampilkan ukuran derajat siku pada bagian siku. Pengujian dilakukan dengan dua variabel, yaitu pengujian keberhasilan deteksi dan pengujian perhitungan gerakan *dumbbell biceps curl*. Pengujian dilakukan didalam ruangan dan diluar ruangan. Hasil uji coba mendapatkan sebuah kesimpulan sistem dapat bekerja dengan baik dalam tingkat cahaya tinggi dan posisi tubuh menghadap ke kamera.

Kata Kunci: Sistem, Pendeteksi, Penghitung, *Dumbbell*, *Biceps curl*, Mediapipe, OpenCV

PENDAHULUAN

Olahraga adalah salah satu kegiatan untuk meningkatkan kualitas kesehatan dan bentuk tubuh manusia. Olahraga merupakan hal penting dalam sehari-hari karena banyak manfaat yang diperoleh melalui kegiatan olahraga. Masyarakat pada umumnya berolahraga agar mengeluarkan keringat, menjaga tubuh tetap sehat dan membentuk tubuh yang atletis. Olahraga juga bermanfaat untuk kesehatan tulang, sendi dan otot. Selain itu, olahraga dapat menurunkan tingkat depresi, kecemasan dan meningkatkan mood.

Gerakan dumbbell yang banyak orang lakukan adalah gerakan dumbbell biceps curl dengan mengangkat dan menurunkan dumbbell sesuai repetisi yang disanggupi. Gerakan dumbbell biceps curl yang bertujuan memperkuat otot bisep atau otot lengan bagian depan. Saat melakukan gerakan dumbbell dan menghitung jumlah gerakannya, saat menghitung terkadang tidak ingat sudah berapa kali melakukan gerakannya dan membuat olahraga yang dilakukan tidak optimal.

OpenCV adalah library perangkat lunak bersifat open source yang memiliki lisensi BSD-Licensed product. OpenCV mempunyai banyak jenis algoritma yang sudah teroptimasi dan disediakan untuk memenuhi kebutuhan mengenai computer vision dan machine learning. Selain itu, penggunaan framework Mediapipe yang dikembangkan oleh perusahaan Google dapat digunakan untuk teknologi computer vision. Berdasarkan penelitian Halder dan Tayade, Mediapipe dapat digunakan secara efisien sebagai alat untuk mendeteksi gerakan tangan yang rumit secara tepat. Fitur landmark dari mediapipe sangat cocok untuk mendeteksi gerakan tangan yang akan digunakan pada penulisan ini.[1]

Dikarenakan permasalahan tersebut, penulis menggunakan library OpenCV dan framework Mediapipe sebagai alat penghitung jumlah gerakan dumbbell biceps curl dan dilakukan penelitian yang berjudul “Sistem Penghitung Jumlah Repetisi Gerakan Dumbbell Biceps Curl Menggunakan Mediapipe dan OpenCV” untuk menghitung gerakan dumbbell secara otomatis dan tidak perlu menghitung lagi secara manual.

METODE PENELITIAN

Metode penelitian yang digunakan pada Penulisan Ilmiah ini adalah menggunakan metode SDLC (Software Development Life Cycle) dengan tahapan berikut:

1. Perencanaan

Tahap ini untuk mencari dan mengumpulkan sumber data yang dibutuhkan untuk membuat sistem jumlah penghitungan repetisi gerakan *dumbbell biceps curl*.

2. Analisis

Tahap ini menganalisis data yang dibutuhkan pada sistem seperti *library* yang dibutuhkan dan perangkat yang dapat menunjang kerja sistem.

3. Perancangan

Tahap ini melakukan perancangan sistem pendeteksi gerak badan dan sistem perhitungan gerak yang dilakukan.

4. Implementasi

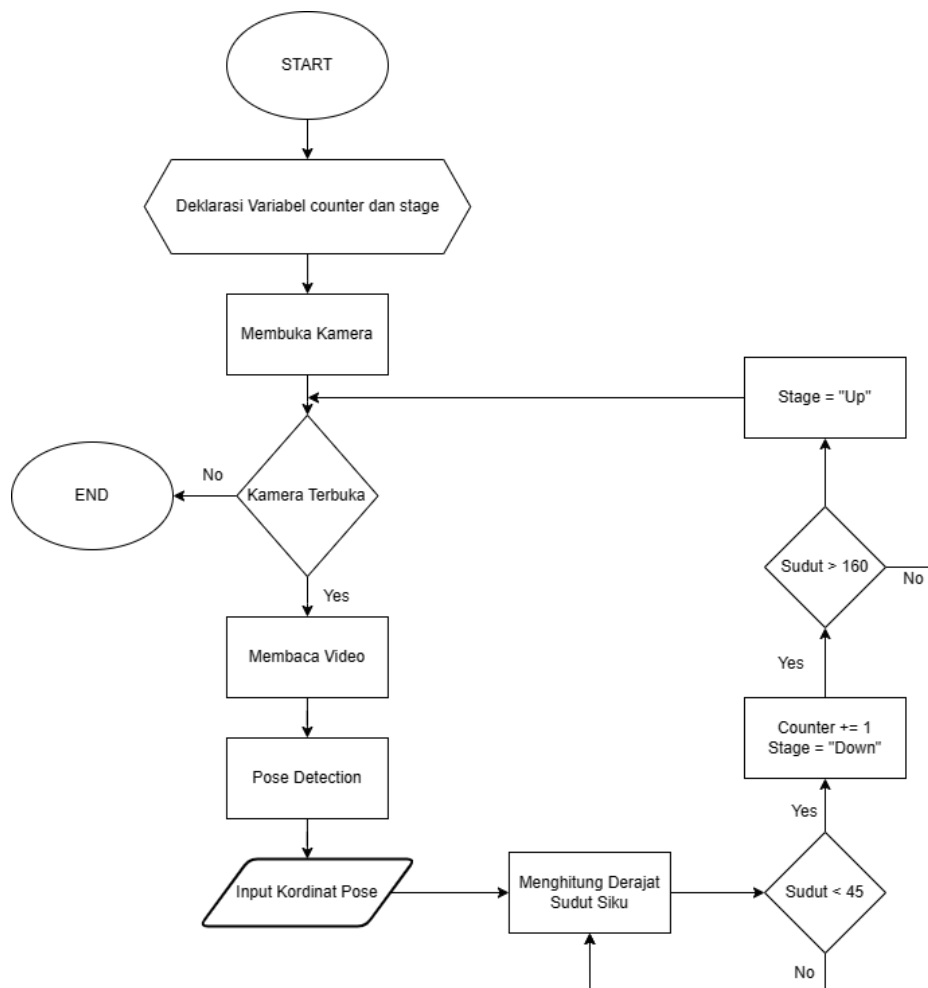
Menggunakan *library* OpenCV untuk mengakses kamera pada laptop. Mendeteksi gerakan lengan dengan membagi menjadi 3 titik, yaitu Bahu, sikut dan pergelangan lengan. Menggunakan library numpy untuk menghitung ukuran derajat pada siku.

5. Uji Coba

Tahap uji coba sistem untuk memastikan apakah sistem berjalan dengan lancar.

HASIL DAN PEMBAHASAN

Sistem ini melakukan pendeteksian bagian lengan secara realtime. Kemudian dari hasil pendeteksian tersebut akan mendapatkan sebuah kordinat dari bagian bahu, siku dan pergelangan tangan. Kordinat tersebut berfungsi untuk menghitung derajat pada bagian siku dan derajat ini akan menjadi sebuah kondisi dimana dalam derajat tertentu akan menghitung gerakan dari dumbbell bicepss curl. Proses pendeteksian menggunakan framework Mediapipe dengan library OpenCV secara realtime.



Gambar 1. Rancangan Diagram Alur

Pada gambar 1 terdapat alur proses sistem. Saat sistem dijalankan akan membuka kamera, kamera tersebut akan menangkap objek tubuh kemudian akan dideteksi pose tubuh dan dihitung derajat sudut sikunya. Apabila sudut siku kurang dari 45 derajat maka akan menghitung sebanyak 1 gearakan dan akan menyimpan instruksi selanjutnya, jika sudut siku lebih dari 160 derajat maka akan menyimpan instruksi selanjutnya. Sistem akan berhenti berjalan jika kamera dimatikan.

1. Instalasi Framework Mediapipe

```
In [1]: !pip install mediapipe opencv-python

Requirement already satisfied: mediapipe in c:\users\mahesa\appdata\roaming\python\python39\site-packages (0.9.0.1)
Requirement already satisfied: opencv-python in c:\users\mahesa\appdata\roaming\python\python39\site-packages (4.6.0.66)
Requirement already satisfied: absl-py in c:\users\mahesa\appdata\roaming\python\python39\site-packages (from mediapipe) (1.3.0)
Requirement already satisfied: attrs>=19.1.0 in c:\users\mahesa\appdata\roaming\python\python39\site-packages (from mediapipe) (22.2.0)
Requirement already satisfied: flatbuffers>=2.0 in c:\users\mahesa\appdata\roaming\python\python39\site-packages (from mediapipe) (22.12.6)
Requirement already satisfied: matplotlib in c:\users\mahesa\appdata\roaming\python\python39\site-packages (from mediapipe) (3.6.2)
Requirement already satisfied: numpy in c:\users\mahesa\appdata\roaming\python\python39\site-packages (from mediapipe) (1.24.0)
Requirement already satisfied: opencv-contrib-python in c:\users\mahesa\appdata\roaming\python\python39\site-packages (from mediapipe) (4.6.0.66)
```

Gambar 2. Instalasi *Framework* Mediapipe

Melakukan instalasi *framework* mediapipe pada jupyter *notebook* yang akan digunakan untuk membuat sistem.

2. Mengimport Framework dan Library

```
1 import cv2
2 import mediapipe as mp
3 import numpy as np
4 mp_drawing = mp.solutions.drawing_utils
5 mp_pose = mp.solutions.pose
```

Gambar 3. Import Library

Untuk kebutuhan sistem penulis mengimport beberapa *library* yang dibutuhkan seperti OpenCV dan numpy. Serta *framework* mediapipe yang telah diinstal sebelumnya.

3. Menghitung Sudut Siku

```
1 def calculate_angle(a,b,c):
2     a = np.array(a) #first
3     b = np.array(b) #second
4     c = np.array(c) #third
5
6     radians = np.arctan2(c[1]-b[1], c[0]-b[0]) - np.arctan2(a[1]-b[1], a[0]-b[0])
7     angle = np.abs(radians*180.0/np.pi)
8
9     if angle > 180.0:
10         angle = 360-angle
11
12     return angle
```

Gambar 4. Function Menghitung Sudut

Dibuat sebuah *function* yang digunakan untuk menghitung derajat sudut pada siku. *Function* ini menerima 3 parameter dimana parameter tersebut sebuah kordinat yang bertipe array. Menggunakan beberapa fungsi dari *library* numpy yaitu np.arctan2 dan np.abs. Apabila sudut yang dihasilkan lebih dari 180 derajat maka 360 derajat dikurang dengan sudut yang lebih dari 180 derajat.

4. Setup Mediapipe

```
1 cap = cv2.VideoCapture(0)
2
3 #Url Counter variables
4 counter_right = 0
5 stage_right = None
6 counter_left = 0
7 stage_left = None
8
9 #setup mediapipe instance
10 with mp_pose.Pose(min_detection_confidence=0.5, min_tracking_confidence=0.5) as pose:
11     while cap.isOpened():
12         ret, frame = cap.read()
13
14         # RECOLOR IMAGE to RGB
15         image = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
16         image.flags.writeable = False
17
18         #make detection
19         result = pose.process(image)
20
21         #RECOLOR back to BGR
22         image.flags.writeable = True
23         image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)
```

Gambar 5. Setup Mediapipe

Mendeklarasikan kamera dengan `cv2.VideoCapture(0)` pada variable `cap`. Membuat beberapa variable seperti `counter_right` dan `counter_left` dengan nilai 0 serta `stage_right` dan `stage_left` bernilai `None`. Melakukan setup mediapipe dengan `with mp_pose.Pose` yang terdapat 2 parameter yaitu `min_detection_confidence=0.5` untuk mengatur ambang batas kepercayaan minimum yang digunakan dalam deteksi objek dimana penulis mengaturnya sebanyak 50% dan `min_tracking_confidence=0.5` untuk mengatur ambang batas kepercayaan minimum yang digunakan dalam deteksi objek dimana penulis mengaturnya sebanyak 50%. Untuk mendapatkan hasil deteksi yang lebih baik, skema warna pada video diubah dari BGR (Blue-Green-Red) menjadi RGB (Red-Green-Blue) dengan fungsi `cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)` dan merubah array gambar menjadi tidak dapat ditulis atau immutable dengan fungsi `flags.writeable`. Barulah video melakukan pendeteksian dengan fungsi `pose.process` dan mengembalikan skema warna dari RGB menjadi BGR serta array gambar menjadi dapat ditulis atau mutable.

5. Mendapatkan Kordinat

```
#extract landmark
try:
    landmarks = result.pose_landmarks.landmark

    #get cordinates
    #RIGHT
    shoulder_right = [landmarks[mp_pose.PoseLandmark.RIGHT_SHOULDER.value].x, landmarks[mp_pose.PoseLandmark.RIGHT_SHOULDER.value].y]
    elbow_right = [landmarks[mp_pose.PoseLandmark.RIGHT_ELBOW.value].x, landmarks[mp_pose.PoseLandmark.RIGHT_ELBOW.value].y]
    wrist_right = [landmarks[mp_pose.PoseLandmark.RIGHT_WRIST.value].x, landmarks[mp_pose.PoseLandmark.RIGHT_WRIST.value].y]
    #LEFT
    shoulder_left = [landmarks[mp_pose.PoseLandmark.LEFT_SHOULDER.value].x, landmarks[mp_pose.PoseLandmark.LEFT_SHOULDER.value].y]
    elbow_left = [landmarks[mp_pose.PoseLandmark.LEFT_ELBOW.value].x, landmarks[mp_pose.PoseLandmark.LEFT_ELBOW.value].y]
    wrist_left = [landmarks[mp_pose.PoseLandmark.LEFT_WRIST.value].x, landmarks[mp_pose.PoseLandmark.LEFT_WRIST.value].y]
```

Gambar 6. Kordinat Lengan

Mendapatkan kordinat dari bagian tubuh, bagian tubuh yang diperlukan adalah bahu, sikut dan pergelangan tangan. Kordinat akan disimpan pada variable yang sudah dibuat dan nilainya berbentuk array berupa hasil dari kordinat x dan y.

6. Kondisi Sudut Siku

```
#curl counter logic
if angle_right > 160:
    stage_right = "down"
if angle_right < 30 and stage_right == "down":
    stage_right="up"
    counter_right += 1
if angle_left > 160:
    stage_left = "down"
if angle_left < 30 and stage_left == "down":
    stage_left = "up"
    counter_left += 1
```

Gambar 7. Kondisi Perhitungan

Membuat curl counter logic yaitu logika perhitungan untuk menghitung jumlah repetisi gerakan dumbbell biceps curl. Menggunakan statement percabangan if dengan kondisi dari variable angle_right atau angle left memenuhi ukuran sudut yang ditentukan untuk menjalankan statement tertentu. Jika ukuran sudut lebih besar dari 160 derajat maka variable stage_right atau stage_left akan bernilai “DOWN”. Jika sudut lebih kecil 30 derajat dan stage_right atau stage_left bernilai “DOWN” maka counter_right atau counter_left bertambah jumlahnya sebanyak 1 nilai dan stage_right atau stage_left bernilai “UP”.

7. Merancang Tampilan Sistem

```
#calculate angle
angle_right = calculate_angle(shoulder_right, elbow_right, wrist_right)
angle_left = calculate_angle(shoulder_left, elbow_left, wrist_left)

#visualize
cv2.putText(image, str(angle_right),
            tuple(np.multiply(elbow_right,[640, 480]).astype(int)),
            cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255,255,255), 2, cv2.LINE_AA
            )

#render curl counter
#setup status box
cv2.rectangle(image,(0,0),(325,73), (245,117,16),-1)
cv2.rectangle(image,(325,0),(650,73), (212,41,58),-1)

#RIGHT
#rep data right
cv2.putText(image, "REPS &", (15,12),
            cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0,0,0), 1, cv2.LINE_AA)
cv2.putText(image, str(counter_right), (10,60),
            cv2.FONT_HERSHEY_SIMPLEX, 2, (255, 255, 255), 2, cv2.LINE_AA)
#stage data right
cv2.putText(image, "STAGE RIGHT", (90,12),
            cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0,0,0), 1, cv2.LINE_AA)
cv2.putText(image, stage_right, (90,60),
            cv2.FONT_HERSHEY_SIMPLEX, 2, (255, 255, 255), 2, cv2.LINE_AA)

#render detections
mp_drawing.draw_landmarks(image, result.pose_landmarks, mp_pose.POSE_CONNECTIONS,
                          mp_drawing.DrawingSpec(color=(245,117,66), thickness=2, circle_radius=2),
                          mp_drawing.DrawingSpec(color=(245,66,280), thickness=2, circle_radius=2)
                          )
```

Gambar 8. Rancangan Tampilan Sistem

Menghitung sudut derajat siku menggunakan fungsi yang telah dibuat dan menampilkan ukurannya pada layar. Membuat sebuah kotak dan text untuk mempresentasikan jumlah perhitungan gerakan.

8. Tombol Pengatur Sistem

```
# Menangkap tombol keyboard
key = cv2.waitKey(1)

# Jika tombol 'r' atau 'l' ditekan, reset perhitungan
if key == ord('r'):
    counter_right = 0
    stage_right = None
if key == ord('l'):
    counter_left = 0
    stage_left = None

if cv2.waitKey(10) & 0xFF == ord('q'):
    break
```

Gambar 9. Tombol Pengatur Sistem

Membuat tombol pada keyboard bekerja pada program untuk mereset jumlah perhitungan repetisi. Mendeklarasikan variable key yang akan menyimpan fungsi untuk mengambil input tombol. Pada percabangan if terdapat kondisi untuk mereset perhitungan gerakan yaitu jika menekan tombol “r” atau “l” maka stage_right atau stage_left bernilai None dan counter_right atau counter_left kembali bernilai 0. Pada percabangan if dengan kondisi `cv2.waitKey(10) & 0xFF == ord('q')` serta statement break untuk memberhentikan sistem secara manual dengan menekan tombol “q”.

9. Implementasi



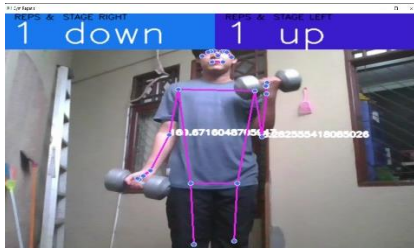
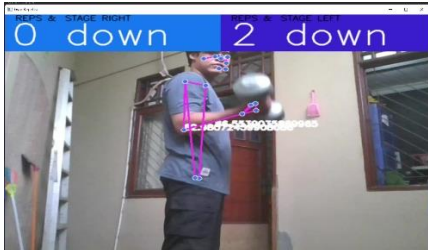

Gambar 9. Tombol Pengatur Sistem

Pada implementasi, akan menampilkan sistem yang sudah dijalankan dengan tampilan awal yang telah dibuat. Yang akan terlihat pada *frame* adalah 2 kotak berwarna biru dan ungu untuk menghitung jumlah gerakan repetisi *dumbbell biceps curl*, ukuran sudut pada bagian siku di kedua sisi tangan dan menampilkan *pose landmark*.



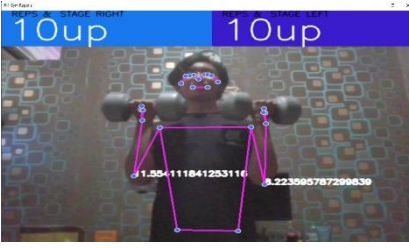
10. Hasil Uji Coba

Pada pengujian sistem penghitung repetisi gerakan dumbbell biceps curl. bahan uji coba yaitu lengan. Uji coba dilakukan untuk mengetahui apakah sistem perhitungan dari subjek akan berhasil secara optimal atau tidak. Lengan yang digunakan adalah bagian kanan dan bagian kiri. Jumlah gerakan yang dilakukan sebanyak 10 gerakan setiap skenario yang diuji cobakan. Hasil uji coba ditampilkan dalam tabel yang memperlihatkan data hasil uji coba. Pada uji coba sistem terdapat beberapa tingkat cahaya yaitu tinggi, sedang dan rendah. Setiap tingkatan cahaya terdapat 3 skenario yang akan diuji cobakan.

Tabel 1. Tabel uji coba dengan tingkat cahaya tinggi

No	Skenario	Tampilan Percobaan	Jumlah Gerakan Terdeteksi	Keterangan
1	1		10 gerakan untuk ke dua tangan	Sistem berjalan dengan baik yaitu mampu mendeteksi pose dan menghitung jumlah gerakan.
2	2		10 gerakan untuk ke dua tangan	Sistem berjalan dengan baik dalam menghitung gerakan tetapi untuk mendeteksi pose terdapat sedikit masalah dalam keakuratan.
3	3		10 gerakan untuk ke dua tangan	Sistem berjalan dengan baik yaitu mampu mendeteksi pose dan menghitung jumlah gerakan.

Tabel 2. Tabel uji coba dengan tingkat cahaya sedang

No	Skenario	Tampilan Percobaan	Jumlah Gerakan Terdeteksi	Keterangan
1	1		10 gerakan untuk ke dua tangan	Sistem berjalan dengan baik yaitu mampu mendeteksi pose dan menghitung jumlah gerakan.
2	2		10 gerakan untuk tangan kiri dan 20 untuk tangan kanan	Sistem berjalan dengan kurang baik dalam menghitung gerakan dan mendeteksi pose tubuh.
3	3		10 gerakan untuk ke dua tangan	Sistem berjalan dengan baik yaitu mampu mendeteksi pose dan menghitung jumlah gerakan.

Tabel 3. Tabel uji coba dengan tingkat cahaya rendah

No	Skenario	Tampilan Percobaan	Jumlah Gerakan Terdeteksi	Keterangan
1	1		10 gerakan untuk tangan kanan dan 30 untuk tangan kiri	Sistem berjalan kurang baik yaitu sulit untuk mendeteksi dan menghitung jumlah gerakan.

No	Skenario	Tampilan Percobaan	Jumlah Gerakan Terdeteksi	Keterangan
2	2		10 gerakan untuk tangan kiri dan 13 untuk tangan kanan	Sistem berjalan kurang baik yaitu sulit untuk mendeteksi dan menghitung jumlah gerakan.
3	3		10 gerakan untuk ke dua tangan	Sistem berjalan dengan baik yaitu mampu mendeteksi pose dan menghitung jumlah gerakan.

Sistem penghitung jumlah repetisi gerakan dumbbell biceps curl menggunakan Mediapipe dan OpenCV telah diuji dengan berbagai skenario uji. Berdasarkan Tabel 3.1, tingkat cahaya tinggi merupakan hasil uji coba yang paling optimal dari pendeteksian pose tubuh dan perhitungan jumlah gerakan dumbbell biceps curl. Hal ini disebabkan oleh tingkat cahaya yang tinggi yang memperlihatkan pose tubuh dengan jelas dan dapat dideteksi dengan baik. Pada Tabel 3.2, tingkat cahaya sedang mendapatkan hasil yang optimum pada skenario 1 dan 3 sedangkan skenario 2 terdapat sedikit masalah dalam pendeteksian pose tubuh yang menyebabkan perhitungan gerakan menjadi tidak optimal. Dan Tabel 3.3, tingkat cahaya rendah menghasilkan kerja sistem penghitung repetisi gerakan dumbbell biceps curl menjadi tidak optimal yang menyebabkan pendeteksian pose tubuh dan penghitungan gerakan dumbbell biceps curl tidak berjalan baik.

KESIMPULAN DA SARAN

Hasil pengujian sistem ini berhasil mendeteksi dan menghitung jumlah gerakan dumbbell bicep curl dalam tiga tingkatan cahaya, yaitu tinggi, sedang dan rendah serta tiga skenario yaitu skenario 1 dilakukan dengan posisi subjek berhadapan lurus dengan kamera dan

bergantian tangan setiap satu repetisi, skenario 2 dilakukan dengan posisi subjek menghadap kesamping dari posisi kamera dengan bergantian tangan setiap satu set, dan skenario 3 dilakukan dengan posisi subjek berhadapan lurus dengan kamera dan gerakan dumbbell dilakukan dengan kedua tangan secara bersamaan.

Kondisi dengan tingkat cahaya yang tinggi merupakan kondisi yang optimum untuk menjalankan sistem penghitung jumlah repetisi gerakan dumbbell bicep curl. Posisi tubuh yang paling baik dalam pendeteksian dan perhitungan gerakan dumbbell bicep curl adalah menghadap ke kamera. Hal ini disebabkan pada posisi ini pose tubuh dapat terlihat dengan jelas sehingga sistem dapat mendeteksi dan menghitung gerakan dengan baik.

Sistem ini tidak tertutup kemungkinan untuk menggunakan sebuah dataset untuk mengoptimalkan kerja sistem, menambahkan jenis gerakan yang dapat dihitung dan diimplementasikan ke dalam android maupun sebuah website.

DAFTAR PUSTAKA

- [1] "Artikel Jurnal 07.Nurhaliza Khesya (0305201021)".
- [2] A. Mordvintsev, "OpenCV-Python Tutorials Documentation Release 1," 2017.
- [3] A. P. Ismail, F. A. A. Aziz, N. M. Kasim, dan K. Daud, "Hand gesture recognition on python and opencv," *IOP Conf Ser Mater Sci Eng*, vol. 1045, no. 1, hlm. 012043, Feb 2021, doi: 10.1088/1757-899x/1045/1/012043.
- [4] A. Yusuf, A. Sahrul Jahrir, dan S.-Y. Makassar, "Pengaruh Latihan Bicep curl dan Preacher curl Terhadap Kemampuan Tangkapan Satu Kaki Olahraga Gulat Mahasiswa STKIP YPUP Makassar," *Jendela Olahraga*, vol. 05, no. 1, hlm. 10–20, 2020.
- [5] C. Lugaresi dkk., "MediaPipe: A Framework for Building Perception Pipelines," Jun 2019, [Daring]. Tersedia pada: <http://arxiv.org/abs/1906.08172>
- [6] I. Yuniarto dan K. Adhiyarta, "JURNAL REVIEW: PERBANDINGAN SISTEM OPERASI LINUX DENGAN SISTEM OPERASI WINDOWS," *Jupiter: Journal of Computer & Information Technology*, vol. 1, no. 1, hlm. 1–7, Jul 2021, doi: 10.53990/cist.v1i1.77.
- [7] Joseph. Howse, *OpenCV Computer Vision with Python*. Packt Publishing, 2013.
- [8] P. Kode Program Dan Simulasi, "LAPORAN PENELITIAN."
- [9] S. Suwarno dan K. Kevin, "Analysis of Face Recognition Algorithm: Dlib and OpenCV," *JOURNAL OF INFORMATICS AND TELECOMMUNICATION ENGINEERING*, vol. 4, no. 1, hlm. 173–184, Jul 2020, doi: 10.31289/jite.v4i1.3865.
- [10] S. U. Rahman, Z. Afroze, dan M. Tareq, "Hand Gesture Recognition Techniques For Human Computer Interaction Using OpenCv," *International Journal of Scientific and Research Publications*, vol. 4, no. 12, 2014, [Daring]. Tersedia pada: www.ijsrp.org.

- [11] Y. Kwon dan D. Kim, "Real-Time Workout Posture Correction using OpenCV and MediaPipe," *The Journal of Korean Institute of Information Technology*, vol. 20, no. 1, hlm. 199–208, Jan 2022, doi: 10.14801/jkiit.2022.20.1.199.