

Penerapan Estimasi Pose dengan Model MoveNet untuk Klasifikasi Posisi Gerakan Deadlift Menggunakan XGBoost

¹Kevin Ragil Krisna Dyansyah, ²Ahmad Bagus Setiawan, ³Patmi Kasih

^{1, 2, 3} Teknik Informatika, Universitas Nusantara PGRI Kediri,

E-mail: ¹kevinragil768@gmail.com, ²ahmadbagus@unpkediri.ac.id, ³fatkasih@gmail.com

Penulis Korespondens : Kevin Ragil Krisna Dyansyah

Abstrak—Klasifikasi fase gerakan pada latihan deadlift penting untuk menghindari cedera akibat postur yang salah. Penelitian ini mengembangkan sistem klasifikasi posisi “up” dan “down” menggunakan estimasi pose berbasis *MoveNet Lightning* dan algoritma *XGBoost*. Sistem mendekripsi 17 keypoint tubuh secara real-time melalui browser dan menghitung tiga sudut utama (pinggul, lutut, punggung) sebagai fitur klasifikasi. Data sudut tersebut diberi label otomatis dan dilatih menggunakan *XGBoost* dengan akurasi uji mencapai 98,51%. Analisis menunjukkan bahwa sudut lutut memiliki kontribusi tertinggi terhadap hasil klasifikasi. Sistem berjalan sepenuhnya di sisi klien menggunakan React.js dan TensorFlow.js, memungkinkan klasifikasi real-time tanpa backend. Pendekatan ini menunjukkan bahwa kombinasi model ringan dan algoritma efisien dapat menghasilkan sistem pelatihan digital yang akurat, ringan, dan portabel.

Kata Kunci— Deadlift, Estimasi Pose, MoveNet, Machine Learning, XGBoost

Abstract— *Classifying movement phases in deadlift exercises is essential to prevent injuries caused by incorrect posture. This study develops a classification system for “up” and “down” positions using pose estimation with MoveNet Lightning and the XGBoost algorithm. The system detects 17 body keypoints in real-time via browser and calculates three primary joint angles (hip, knee, back) as classification features. These angle data are automatically labeled and trained using XGBoost, achieving a test accuracy of 98.51%. Analysis shows that knee angle contributes the most to classification performance. The system runs entirely on the client side using React.js and TensorFlow.js, enabling real-time classification without backend processing. This approach demonstrates that combining lightweight pose estimation models and efficient machine learning algorithms can result in an accurate, lightweight, and portable digital coaching system.*

Keywords— Deadlift, Machine Learning, MoveNet, Pose Estimation, XGBoost

This is an open access article under the CC BY-SA License.



I. PENDAHULUAN

Analisis gerakan olahraga penting untuk meningkatkan performa sekaligus mencegah cedera, terutama pada cabang seperti powerlifting yang mencakup gerakan kompleks seperti deadlift. Kesalahan postur pada deadlift berisiko tinggi terhadap cedera punggung bawah, sehingga dibutuhkan sistem otomatis yang mampu memantau dan mengklasifikasi posisi tubuh secara real-time. Teknologi *pose estimation* seperti *MoveNet Lightning* memungkinkan

deteksi 17 titik tubuh secara cepat (>30 FPS) [1], dan telah banyak digunakan dalam analisis gerakan olahraga dan fisioterapi [2].

Penelitian sebelumnya telah menunjukkan bahwa data pose efektif dalam mengklasifikasi gerakan. Misalnya, Çelikel dan Özkan menggunakan fitur dari pose estimation untuk klasifikasi gerakan powerlifting dengan akurasi tinggi menggunakan CNN [3], sedangkan Arrowsmith et al. berhasil membedakan latihan fisioterapi dengan akurasi hingga 99% [4]. Beberapa studi juga menyarankan pemanfaatan sudut sendi sebagai fitur klasifikasi yang efisien [5]. Di sisi lain, metode seperti XGBoost menawarkan alternatif klasifikasi yang ringan dan akurat tanpa memerlukan sumber daya besar seperti pada CNN [6]. Metode XGBoost juga menawarkan keunggulan signifikan baik dari sisi kecepatan maupun akurasi dalam memproses dataset kompleks dan berskala besar[7].

Namun, masih sedikit studi yang secara khusus membahas klasifikasi fase gerakan deadlift (naik dan turun) dengan pendekatan ringan yang cocok untuk sistem real-time. Belum banyak juga yang menggabungkan model efisien seperti MoveNet Lightning dengan algoritma XGBoost. Penelitian ini bertujuan mengisi celah tersebut dengan mengembangkan dan menguji sistem klasifikasi fase deadlift berbasis data sudut tubuh menggunakan MoveNet dan XGBoost, serta mengevaluasi akurasinya dalam konteks implementasi real-time.

II. METODE

Penelitian ini menggunakan pendekatan metodologi Waterfall karena seluruh proses dikembangkan secara berurutan dan terstruktur, mulai dari analisis kebutuhan hingga implementasi sistem. Setiap tahapan diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya. Model ini cocok digunakan karena sistem bersifat terukur, spesifik, dan tidak memerlukan iterasi kompleks antar modul.

A. Analisis Kebutuhan

Penelitian ini membutuhkan kombinasi perangkat keras, perangkat lunak, serta data input yang mendukung proses estimasi pose dan klasifikasi gerakan deadlift secara real-time. Data utama berupa video gerakan deadlift dengan tampilan frontal diunduh dari platform YouTube dalam format MP4. Video ini digunakan sebagai sumber frame untuk ekstraksi pose tubuh.

Sistem dirancang untuk berjalan sepenuhnya di sisi klien (browser) menggunakan TensorFlow.js dengan model MoveNet Lightning, yang mampu mendeteksi 17 titik tubuh secara cepat lebih dari 30 frame per detik (FPS) [1]. Setiap frame diproses secara real-time, dan koordinat keypoint tubuh digunakan untuk menghitung tiga sudut utama dalam gerakan deadlift, yaitu sudut pinggul (hipAngle), sudut lutut (kneeAngle), dan sudut punggung (backAngle).

Untuk menghitung sudut antara tiga titik tubuh (A, B, dan C), dua vektor dibentuk terlebih dahulu, yaitu vektor \overrightarrow{BA} dan \overrightarrow{BC} , seperti pada Persamaan (1):

$$\overrightarrow{BA}=(x_1-x_3, y_1-y_3), \overrightarrow{BC}=(x_2-x_3, y_2-y_3) \quad (1)$$

Sudut θ di titik B dihitung menggunakan rumus dot product dua vektor, yang ditunjukkan pada Persamaan (2):

$$\theta = 180^\circ - \cos^{-1} \left(\frac{(x_1-x_3)(x_2-x_3)+(y_1-y_3)(y_2-y_3)}{\sqrt{(x_1-x_3)^2+(y_1-y_3)^2} \cdot \sqrt{(x_2-x_3)^2+(y_2-y_3)^2}} \right) \quad (2)$$

Persamaan ini digunakan untuk menghitung sudut sendi seperti lutut dan pinggul, dengan titik B sebagai pusat sudut. Nilai sudut hasil perhitungan dikurangi dari 180° untuk menyesuaikan orientasi sudut yang relevan secara biomekanik.

Untuk sudut punggung (backAngle), digunakan pendekatan vektor terhadap garis vertikal buatan. Sebuah titik imajiner dibuat 100 piksel di atas pinggul pada sumbu y sebagai acuan garis tegak lurus. Titik tersebut didefinisikan sebagai berikut (3):

$$P_{\text{vertical}} = (X_{\text{hip}}, Y_{\text{hip}} - 100) \quad (3)$$

Sudut punggung kemudian dihitung sebagai sudut antara bahu, pinggul, dan titik vertikal ini, menggunakan rumus yang sama seperti Persamaan (2).

Rumus ini diterapkan dalam program menggunakan fungsi calculateAngle() di JavaScript, dan dikombinasikan dengan fungsi calculateBackAngle() yang membentuk referensi garis vertikal dari pinggul untuk menghitung sudut punggung. Nilai sudut dari setiap frame dicetak ke konsol browser dan dieksport dalam format CSV. Data ini kemudian diproses dalam lingkungan Python menggunakan pustaka Pandas, NumPy, Matplotlib, dan Seaborn untuk eksplorasi data. Untuk pelatihan dan evaluasi model klasifikasi, digunakan algoritma XGBoost, serta modul Scikit-learn untuk pembagian data dan perhitungan metrik performa. Model akhir disimpan dengan Joblib dalam format .joblib agar dapat digunakan kembali dalam sistem klasifikasi rule-based di web.

B. Pengolahan Citra

Proses labeling gerakan dilakukan secara otomatis berdasarkan nilai sudut panggul (hipAngle). Frame dengan hipAngle < 145 diklasifikasikan sebagai posisi down, sedangkan hipAngle 145 sebagai up. Aturan ini diimplementasikan dalam skrip JavaScript secara langsung saat proses estimasi pose berlangsung, sehingga seluruh frame diberi label secara otomatis tanpa anotasi manual. Label dan nilai sudut dicatat ke file CSV.

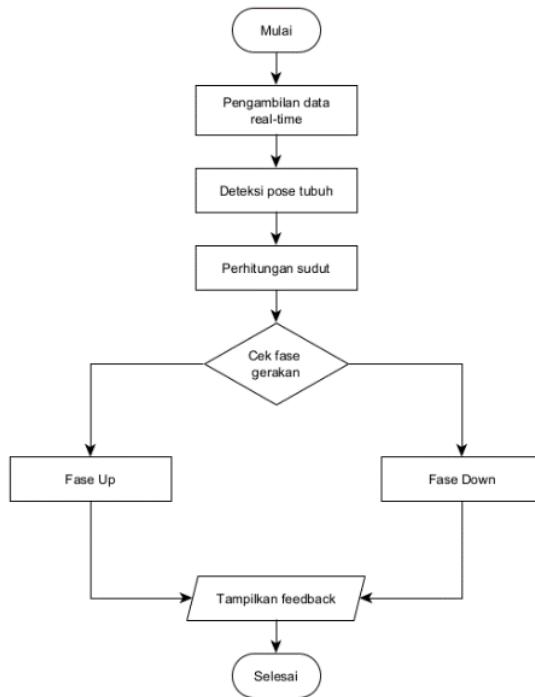
Data CSV yang memuat fitur sudut dan label dibaca dalam Python dan dibagi menggunakan fungsi train_test_split dari Scikit-learn dengan rasio 80:20. Desain eksperimen dilakukan dengan pendekatan supervised learning dua kelas. Algoritma XGBoost (XGBClassifier) digunakan untuk melatih model. XGBoost merupakan algoritma gradient boosting yang efisien dan terbukti kompetitif dalam berbagai kompetisi data mining [8]. XGBoost termasuk metode yang sangat akurat dan cepat dalam pembelajaran [9]. Penggunaan XGBoost terbukti efektif, dan sesuai dengan studi sebelumnya yang melaporkan akurasi tinggi pada klasifikasi pose olahraga menggunakan metode serupa [10].

Evaluasi dilakukan dengan menghitung akurasi, precision, recall, F1-score, serta confusion matrix. Visualisasi korelasi fitur dan matriks kebingungan dibuat dengan Matplotlib dan Seaborn. Teknik filtering juga diterapkan untuk membuang frame dengan keypoint confidence score < 0.5 guna menjaga kualitas data input.

C. Tahapan Implementasi Sistem

Sistem akhir diimplementasikan dalam antarmuka web berbasis React.js yang terintegrasi dengan TensorFlow.js untuk estimasi pose secara real-time. Sistem ini sepenuhnya berjalan di sisi klien (client-side), tanpa membutuhkan koneksi server, sehingga

responsif dan efisien. Antarmuka menampilkan tangkapan kamera secara langsung, overlay skeleton pose, status fase gerakan, serta hitungan repetisi secara otomatis berdasarkan transisi up/down.



Gambar 1. Flowchart Sistem

Flowchart diatas menggambarkan tahapan utama dalam implementasi sistem klasifikasi fase deadlift. Proses dimulai dengan input jumlah repetisi oleh pengguna, dilanjutkan dengan pengambilan data secara real-time melalui kamera. Sistem kemudian mendeteksi pose tubuh, menghitung sudut, dan mengklasifikasi fase gerakan sebagai "up" atau "down". Berdasarkan klasifikasi tersebut, sistem akan menampilkan umpan balik visual kepada pengguna hingga jumlah repetisi terpenuhi.

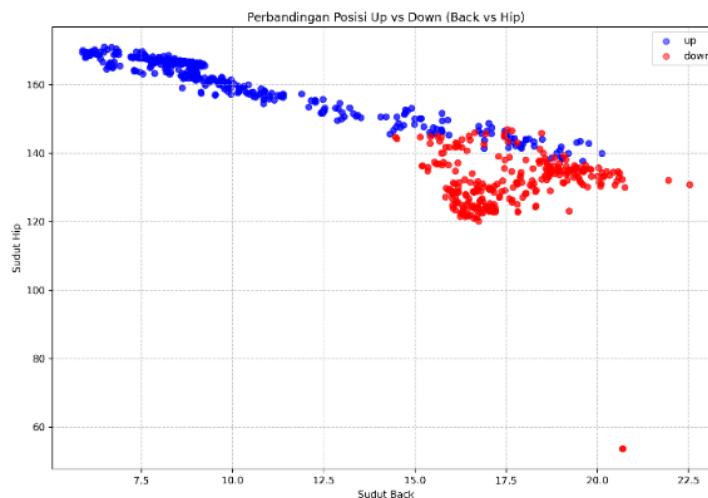
III. HASIL DAN PEMBAHASAN

A. Visualisasi Distribusi Data

Untuk menganalisis pola distribusi gerakan deadlift berdasarkan fitur sudut tubuh, dilakukan visualisasi dua dimensi terhadap kombinasi fitur utama, yaitu sudut pinggul (hipAngle), sudut lutut (kneeAngle), dan sudut punggung (backAngle). Visualisasi ini bertujuan untuk mengobservasi pemisahan (separabilitas) antara dua kelas: posisi "up" (naik) dan posisi "down" (turun), yang kemudian menjadi dasar dalam pelatihan model klasifikasi.

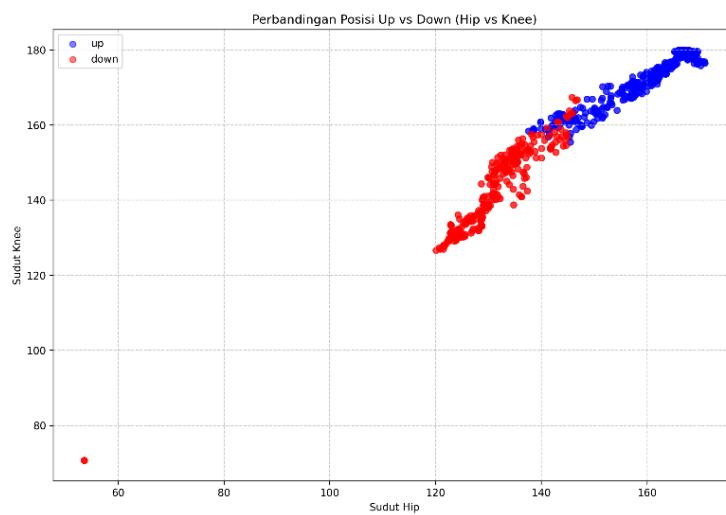
Gambar 2 memperlihatkan plot sebar antara sudut back (sumbu X) dan hip (sumbu Y). Warna biru mewakili posisi "up" sedangkan merah untuk posisi "down". Terlihat bahwa posisi "up" terkonsentrasi pada wilayah dengan sudut back rendah (sekitar 5–12 derajat) dan sudut hip tinggi (sekitar 155–175 derajat). Sebaliknya, posisi "down" memiliki kecenderungan sebaliknya: sudut back lebih tinggi (sekitar 15–25 derajat), dan sudut hip lebih rendah (sekitar 120–145 derajat). Pola ini menunjukkan bahwa postur tubuh saat menurunkan beban

mengakibatkan sudut punggung membuka (lebih tegak) dan panggul menekuk lebih dalam. Visualisasi ini mempertegas bahwa kombinasi sudut back dan hip mampu membedakan fase gerakan secara efektif.



Gambar 2. Perbandingan Posisi Up vs Down Berdasarkan Sudut Back dan Hip

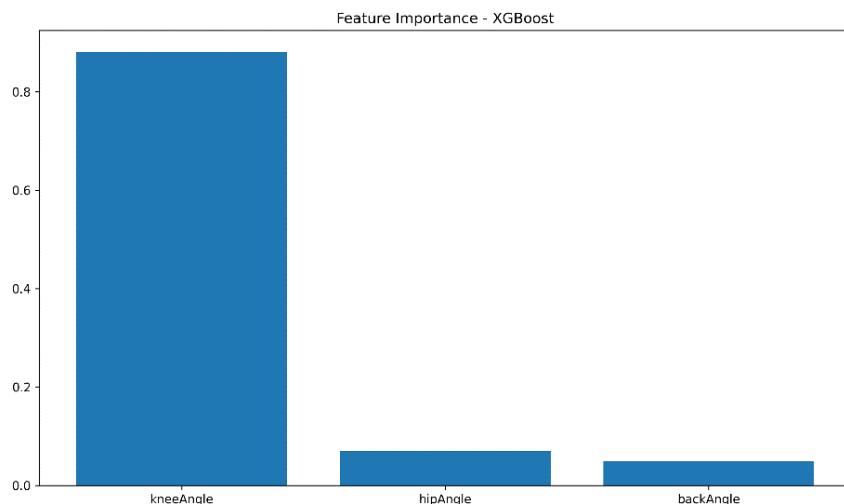
Gambar 3 menunjukkan sebaran antara sudut hip (sumbu X) dan knee (sumbu Y). Sama seperti sebelumnya, warna biru menunjukkan posisi “up”, dan merah untuk posisi “down”. Terlihat bahwa posisi “up” memiliki sudut knee yang lebih lurus (sekitar 165–180 derajat) dan sudut hip yang tinggi. Di sisi lain, posisi “down” memiliki sudut knee yang lebih menekuk (sekitar 135–155 derajat) bersamaan dengan sudut hip yang lebih kecil. Pola ini konsisten dengan biomekanika gerakan deadlift, di mana saat fase turun, lutut akan lebih menekuk dan pinggul ikut menurun untuk menyesuaikan posisi beban di dekat lantai. Grafik ini juga menunjukkan clustering yang jelas antar kelas, menandakan bahwa pemisahan antara posisi “up” dan “down” secara geometrik dapat dijelaskan secara visual dan logis.



Gambar 3. Perbandingan Posisi Up vs Down berdasarkan Sudut Hip dan Knee

C. Analisis Fature Importance

Analisis feature importance pada model XGBoost menunjukkan bahwa sudut lutut (kneeAngle) memiliki kontribusi paling besar dalam klasifikasi fase gerakan deadlift, dengan nilai importance sekitar 0.86. Fitur ini terbukti paling konsisten dalam membedakan posisi "up" dan "down", mengingat sudut lutut berubah drastis antara posisi berdiri dan menekuk. Di sisi lain, sudut pinggul (hipAngle) menyumbang sekitar 0.10, sedangkan sudut punggung (backAngle) hanya sekitar 0.04.



Gambar 4. Feature Importance

Temuan ini menyoroti bahwa meskipun pinggul sering dianggap sebagai pusat gerakan deadlift, secara statistik lutut justru lebih informatif. Perbedaan ini menegaskan keunggulan pendekatan *machine learning* dalam mengungkap pola relevan yang mungkin terlewatkan oleh observasi manual. Fokus pada kneeAngle juga mendukung efisiensi sistem secara komputasi karena mudah dihitung dan stabil secara deteksi, menjadikannya cocok untuk aplikasi *real-time* di browser.

D. Pengujian Sistem Secara Real-time

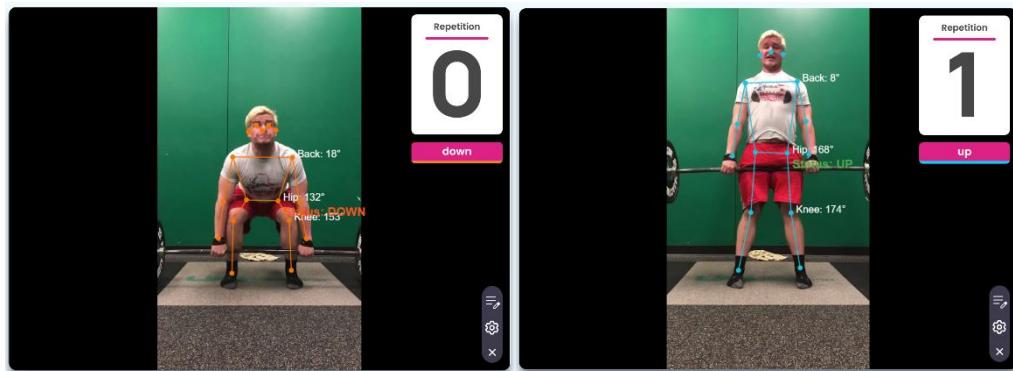
Pengujian dilakukan untuk mengevaluasi kinerja sistem klasifikasi gerakan deadlift secara langsung melalui webcam. Sistem berjalan sepenuhnya di sisi klien menggunakan React.js dan TensorFlow.js, tanpa memerlukan backend atau pemrosesan server eksternal. Deteksi pose dilakukan secara real-time menggunakan model MoveNet Lightning, yang menghasilkan 17 keypoint tubuh dari tangkapan webcam. Dari keypoint ini, sistem menghitung sudut pinggul (hip), lutut (knee), dan punggung (back), lalu mengklasifikasikan fase gerakan menggunakan aturan berbasis threshold hasil pelatihan model XGBoost. Tiga nilai ambang yang digunakan untuk klasifikasi adalah: $\text{hipAngle} \geq 145,89^\circ$, $\text{kneeAngle} \geq 158,07^\circ$, dan $\text{backAngle} \leq 13,95^\circ$. Posisi "down" dikenali jika seluruh kondisi ini terpenuhi, sementara posisi "up" jika salah satu tidak terpenuhi. Fungsi klasifikasi tersebut ditulis dalam JavaScript dan diintegrasikan langsung ke aplikasi untuk mendukung klasifikasi instan secara lokal. Pengujian pertama dilakukan pada video frontal dengan kamera sejajar pandangan mata, dan hasilnya menunjukkan sistem mampu mendeteksi dan mengklasifikasikan fase "up" dan

"down" secara akurat. Gambar berikut menunjukkan implementasi sistem saat pengguna berada pada dua posisi utama:



Gambar 5 . Pengujian Sistem Pada Posisi Kamera Tepat Ditengah

Selanjutnya dilakukan pengujian kedua menggunakan video dengan sudut kamera yang lebih rendah dan lebih dekat ke pengguna. Meskipun terjadi perubahan perspektif, sistem tetap mampu mendeteksi keypoint tubuh secara akurat dan mengklasifikasikan fase gerakan dengan tepat. Gambar berikut menunjukkan hasil pengujian dengan sudut kamera yang berbeda:



Gambar 6. Pengujian Sistem pada posisi kamera lebih dekat dan rendah

Hasil dari dua skenario pengujian ini menunjukkan bahwa sistem tetap konsisten dan tangguh terhadap variasi posisi kamera. Pengguna tetap mendapatkan feedback visual secara real-time, baik pada sudut pandang sejajar maupun rendah. Dengan hanya mengandalkan tiga fitur sudut utama, sistem terbukti andal dan praktis untuk mendeteksi fase deadlift tanpa ketergantungan perangkat keras atau server tambahan.

E. Hasil Evaluasi Klasifikasi dengan XGBoost

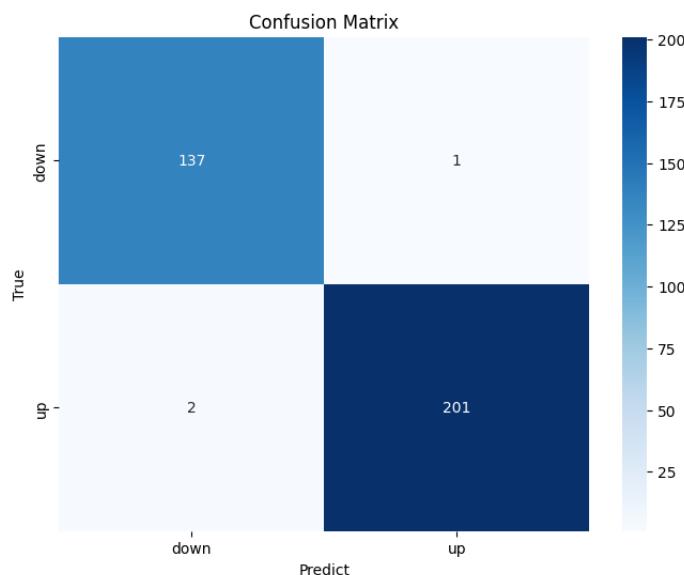
Pada tahapan evaluasi dan klasifikasi dengan XGBoost, pemilihan algoritma ini didasarkan pada kemampuannya dalam menangani data tabular secara efisien, serta keunggulannya dalam hal akurasi dan interpretabilitas fitur. Untuk menjaga proporsi kelas tetap seimbang, dataset dibagi menggunakan teknik *stratified train-test split* dengan rasio 75% data untuk pelatihan dan 25% untuk pengujian. Parameter model disetel secara default, yakni kedalaman maksimum pohon (*max_depth*) sebesar 5, jumlah estimator sebanyak 100, dan laju

pembelajaran (*learning_rate*) sebesar 0.1, dengan jenis objektif *binary:logistic* karena hanya terdapat dua kelas (up dan down).

Tabel 1. Hasil Evaluasi Klasifikasi XGBoost

Kelas	Precision	Recall	F1-Score	Support
Down	0.99	0.99	0.99	138
Up	1.00	0.99	0.99	203

Hasil pelatihan menunjukkan bahwa model mampu mengklasifikasi posisi gerakan deadlift dengan sangat baik. Pada data pengujian, akurasi yang dicapai adalah 98,51%, yang berarti bahwa dari total 341 sampel, hanya terdapat tiga instance yang salah diklasifikasikan. Angka ini mengindikasikan bahwa model memiliki generalisasi tinggi dan sangat minim terhadap kesalahan klasifikasi. Selain akurasi, metrik evaluasi lainnya seperti precision, recall, dan F1-score untuk kedua kelas juga menunjukkan nilai yang sangat tinggi, semuanya di atas 0.98. Hal ini menunjukkan bahwa model tidak hanya dapat mengidentifikasi posisi “up” dan “down” dengan akurat, tetapi juga konsisten dalam menghindari kesalahan prediksi yang merugikan, terutama pada aplikasi real-time di mana ketepatan klasifikasi sangat krusial.



Gambar 7. Confusion Matrix

Untuk menggambarkan detail performa model secara visual, Gambar 4 menampilkan *confusion matrix* hasil klasifikasi pada data uji. Pada gambar tersebut, terlihat bahwa hampir seluruh prediksi berada pada diagonal utama matriks, yang menunjukkan bahwa sebagian besar data berhasil diklasifikasikan secara tepat. Secara spesifik, terdapat 167 instance dari kelas “up” yang diprediksi benar, serta 171 instance dari kelas “down” yang juga diklasifikasikan secara akurat. Hanya terdapat satu kesalahan klasifikasi untuk posisi “up” dan dua untuk posisi “down”, yang merupakan margin kesalahan sangat kecil dari total keseluruhan data pengujian. Visualisasi ini menegaskan bahwa model tidak hanya unggul

secara angka, tetapi juga stabil terhadap distribusi kelas yang seimbang, tanpa bias terhadap salah satu kategori

Penting untuk dicatat bahwa hasil klasifikasi ini diperoleh dengan menggunakan hanya tiga fitur sudut yang dihitung secara sederhana dari koordinat keypoint tubuh. Tidak digunakan fitur visual yang kompleks seperti citra RGB atau keluaran konvolusional dari jaringan saraf dalam (CNN), sehingga menjadikan pendekatan ini jauh lebih ringan secara komputasi dan mudah diimplementasikan di sisi klien (browser). Pendekatan yang efisien ini sangat relevan untuk aplikasi seperti pelatih digital atau sistem koreksi postur otomatis, di mana kecepatan dan akurasi harus berjalan beriringan. Selain itu, hasil pelatihan ini memungkinkan penyusunan aturan klasifikasi berbasis threshold yang dapat diimplementasikan ulang dalam bentuk skrip JavaScript sederhana, sehingga pengguna dapat memanfaatkan fungsi klasifikasi langsung dari perangkat tanpa perlu proses inferensi backend. Hal ini memberikan nilai tambah dari sisi efisiensi, portabilitas, dan kemudahan integrasi ke dalam sistem berbasis web

IV. KESIMPULAN

Penelitian ini berhasil mengembangkan dan menguji sistem klasifikasi fase gerakan deadlift berbasis estimasi pose menggunakan model MoveNet Lightning dan algoritma XGBoost. Dengan memanfaatkan tiga fitur sudut tubuh utama—sudut pinggul, lutut, dan punggung—model klasifikasi mampu mencapai akurasi sebesar 98,51% dalam membedakan posisi “up” dan “down”. Hasil analisis menunjukkan bahwa sudut lutut memiliki kontribusi tertinggi dalam klasifikasi, disusul oleh sudut pinggul dan punggung. Sistem diuji secara real-time melalui kamera webcam tanpa perlu mengunggah video atau melakukan inferensi backend, dan mampu memberikan hasil klasifikasi secara instan di sisi klien melalui antarmuka berbasis React.js dan TensorFlow.js. Implementasi berbasis aturan dari hasil pelatihan model juga memungkinkan sistem berjalan dengan ringan dan responsif pada berbagai perangkat. Pengujian dilakukan pada dua skenario video dengan sudut kamera berbeda (sejajar mata dan lebih rendah), dan hasilnya menunjukkan sistem tetap mampu mengenali fase gerakan secara konsisten dan akurat. Hal ini membuktikan bahwa kombinasi model estimasi pose ringan dan algoritma machine learning efisien dapat digunakan untuk membangun sistem pelatih digital yang portabel, adaptif, dan praktis. Ke depan, pengembangan sistem ini dapat diarahkan pada generalisasi model untuk berbagai bentuk tubuh dan variasi teknik deadlift, serta memperluas sudut pengambilan kamera untuk mengakomodasi deteksi dari sisi samping atau perspektif lainnya. Selain itu, integrasi dengan metrik biomekanik tambahan seperti kecepatan, kestabilan postur, atau simetri gerakan dapat meningkatkan akurasi dan fungsionalitas sistem dalam konteks monitoring latihan yang lebih kompleks.

DAFTAR PUSTAKA

- [1] TensorFlow Hub, “MoveNet: Ultra fast and accurate pose detection model,” <https://www.tensorflow.org/hub/tutorials/movenet>.
- [2] C. Arrowsmith, D. Burns, T. Mak, M. Hardisty, and C. Whyne, “Physiotherapy Exercise Classification with Single-Camera Pose Detection and Machine Learning,” *Sensors*, vol. 23, no. 1, p. 363, Dec. 2022, doi: 10.3390/s23010363.

- [3] A. ÇELİKEL and I. A. OZKAN, “Deep Learning-Based Classification of Powerlifting Movements Using Mediapipe,” *International Journal of Applied Methods in Electronics and Computers*, Sep. 2024, doi: 10.58190/ijamec.2024.107.
- [4] C. Arrowsmith, D. Burns, T. Mak, M. Hardisty, and C. Whyne, “Physiotherapy Exercise Classification with Single-Camera Pose Detection and Machine Learning,” *Sensors*, vol. 23, no. 1, p. 363, Dec. 2022, doi: 10.3390/s23010363.
- [5] C. Mercadal-Baudart *et al.*, “Exercise quantification from single camera view markerless 3D pose estimation,” *Heliyon*, vol. 10, no. 6, p. e27596, Mar. 2024, doi: 10.1016/j.heliyon.2024.e27596.
- [6] B. Liang, W. Qin, and Z. Liao, “A Differential Evolutionary-Based XGBoost for Solving Classification of Physical Fitness Test Data of College Students,” *Mathematics*, vol. 13, no. 9, p. 1405, Apr. 2025, doi: 10.3390/math13091405.
- [7] J. M. Ahn, J. Kim, and K. Kim, “Ensemble Machine Learning of Gradient Boosting (XGBoost, LightGBM, CatBoost) and Attention-Based CNN-LSTM for Harmful Algal Blooms Forecasting,” *Toxins (Basel)*, vol. 15, no. 10, p. 608, Oct. 2023, doi: 10.3390/toxins15100608.
- [8] T. Chen and C. Guestrin, “XGBoost,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA: ACM, Aug. 2016, pp. 785–794. doi: 10.1145/2939672.2939785.
- [9] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, “A Comparative Analysis of XGBoost,” Nov. 2019, doi: 10.1007/s10462-020-09896-5.
- [10] W. Wahid, A. A. AlArfaj, E. A. Alabdulqader, T. Sadiq, H. Rahman, and A. Jalal, “Advanced Human Pose Estimation and Event Classification Using Context-Aware Features and XGBoost Classifier,” *IEEE Access*, vol. 12, pp. 179839–179856, 2024, doi: 10.1109/ACCESS.2024.3498093.