Sensor Applications

OPTIMIZATION OF LOW-COST FLEX SENSOR PERFORMANCE USING MACHINE LEARNING

Carla Zuccarini[1*], Ava Hedayatipour[2**]

[1] Department of Computer Engineering and Computer Science, California State University, Long Beach, CA, 90840, USA
[2] Department of Electrical Engineering, California State University, Long Beach, CA, 90840, USA
[*] Member, IEEE
[**] Senior Member, IEEE

Manuscript received X; revised X; accepted X. Date of publication X; date of current version X.

Abstract—Flex sensors are pivotal in wearable technology, robotics, and gesture recognition due to their affordability and flexibility. However, their performance is often hindered by non-linear behavior, environmental sensitivity, and variability. This paper explores the enhancement of low-cost flex sensor performance through machine learning regression models. A system was designed using an Arduino-based setup for automated data collection, integrating a servo motor for precise angle control and a voltage divider for resistance measurement. Several regression models, including Linear Regression, Polynomial Regression, Random Forest, XGBoost, Gaussian Process Regression (GPR), and a Stacked Ensemble, were evaluated based on Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R^2 scores. Results showed that GPR achieved the highest prediction accuracy with an MAE of 0.82° and R^2 of 0.9981, outperforming traditional methods. These findings demonstrate that applying advanced machine learning techniques significantly enhances the predictive reliability of flex sensors, making them more viable for precision applications in wearable and embedded systems.

Index Terms—Sensors, Flexible Sensor, Arduino, Machine Learning.

I. INTRODUCTION

Sensors are the silent enablers behind modern intelligent systems, translating physical movement into digital insight. Among them, flex sensors—thin, flexible components that change resistance when bent—have emerged as powerful tools for tracking motion and recognizing gestures. cording to the U.S. Census Bureau's 2023 American Community Survey, 44.7 million Americans, or 13.6% of the civilian noninstitutionalized population, reported having a disability [1]. For many individuals with disabilities, such as those with mobility impairments affecting 12.2% of adults [2], gesture recognition systems integrated with assistive devices offer the ability to restore autonomy and improve daily functioning [3]. These technologies provide intuitive interfaces for controlling robotic limbs, enhancing rehabilitation progress, and maintaining motor control for amputees. Furthermore, intelligent sensor networks and IoT solutions facilitate seamless interaction with assistive devices, promoting communication and accessibility [4]. While low-cost flex sensors are accessible and versatile, they suffer from significant inaccuracies due to non-linear responses, sensor drift, and environmental factors. Conversely, high-precision sensors that offer superior accuracy often cost thousands of dollars, creating a substantial barrier for widespread adoption in cost-sensitive applications. Given their potential to enhance human-computer interactions and support people with disabilities, improving flex sensor reliability is critical and machine learning offers a promising solution by modeling complex relationships in sensor data.

Recent studies have demonstrated that machine learning significantly improves the performance of sensor-driven systems. Salman et al. reviewed the application of machine learning in soft robotics, highlighting how Gaussian Process Regression (GPR) and Neural Networks can model the uncertainties inherent in flexible sensor inputs, enhancing robotic adaptability and responsiveness [5]. Akhund et al. integrated flex sensors with Gaussian Mixture Models (GMM) and reinforcement learning in a robotic arm, effectively distinguishing between normal and abnormal hand gestures [6]. Their system showcases the potential for predictive control in assistive and rehabilitation technologies.

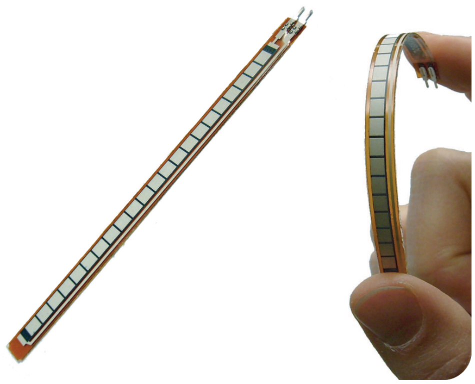
Gesture recognition systems have also been advanced using machine learning algorithms. Panda et al. implemented a hand gesture recognition framework using Random Forest, k-NN, and Support Vector Machines (SVM), achieving high classification accuracies by preprocessing flex sensor data and applying ensemble learning techniques [7]. Likewise, Feng et al. utilized neural networks and transfer learning to address variability across users, significantly improving the cross-user adaptability of wearable gesture recognition devices [8]. These approaches demonstrate the critical role of machine learning in enhancing user-specific accuracy and real-time adaptability in wearable systems.

Further studies have explored multi-modal sensor fusion strategies. Doan and Nguyen combined flex sensors and Inertial Measurement Units (IMUs) to develop a robust gesture recognition platform, using Convolutional Neural Networks (CNNs) for data fusion and classification [9]. This multi-sensor approach reduced noise sensitivity and increased recognition accuracy, indicating that integrating various sensor modalities and advanced machine learning can create highly reliable and adaptive wearable technologies.

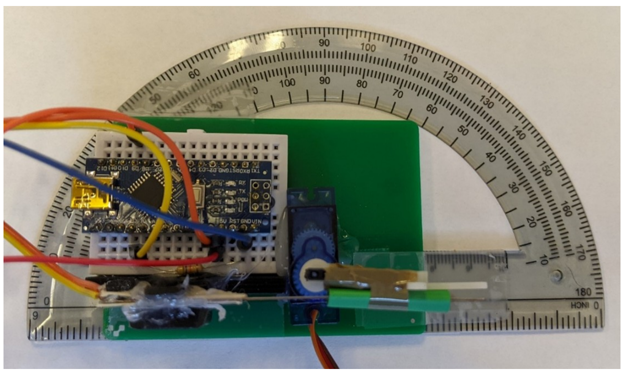
This paper explores the optimization of low-cost flex sensors using supervised learning techniques to improve bending angle prediction accuracy. The novelty of this work lies in the comprehensive evaluation of various machine learning models, the integration of an automated data collection setup with real-time calibration, and the application of Gaussian Process Regression to achieve superior prediction accuracy

Corresponding author: A. Hedayatipour (e-mail: ava.hedayatipour@csulb.edu).

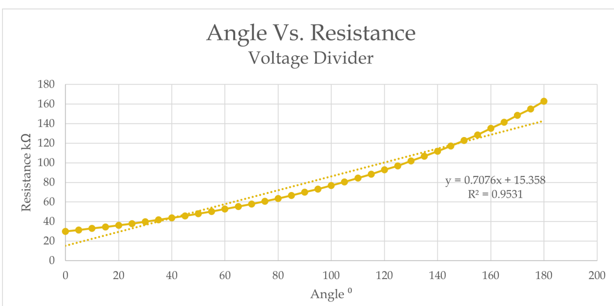
Associate Editor: X
Digital Object Identifier 10.1109/LSENS.2017.0000000



(a)



(b)



(c)

Fig. 1: (a) Sparkfun Flex Sensor [10]; (b) Prototype – Flex sensor attached to the servo with Arduino NANO readout. (c) angle versus resistance data obtained

along with uncertainty estimation.

The remainder of this paper is structured as follows: the next section presents the design of the experimental setup and data acquisition system. Following that, the Results section evaluates the performance of different machine learning models. Finally, the Conclusion summarizes key findings and outlines potential future directions..

II. DESIGN

To enable precise, automated collection of high-resolution flex sensor data for machine learning applications, a custom experimental platform was developed integrating an Arduino microcontroller, servo motor, and signal conditioning circuitry. The experimental setup focused on automating data collection to ensure consistency and precision. A SparkFun flex sensor was connected to an Arduino microcontroller via a voltage divider circuit. A servo motor was employed to bend the flex sensor at precise angles, while real-time resistance data was logged through serial communication (Fig. 1).

The core hardware components included the Arduino Uno, a standard servo motor, a flex sensor (SparkFun SEN-10264), and a 10k Ω fixed resistor forming the voltage divider. The servo motor was programmed to rotate through predefined angles, ensuring repeatability across measurements. The Arduino's analog-to-digital converter (ADC) captured voltage variations corresponding to changes in flex sensor resistance. To further enhance the stability of voltage readings, an operational amplifier was integrated as a voltage buffer. This prevented ADC loading effects and ensured reliable data acquisition. Additionally, data was transmitted via serial communication to a PC where it was logged and stored for machine learning training.

Beyond hardware, careful attention was paid to sensor placement and mechanical design. The flex sensor was mounted on a rigid frame with minimal external disturbances to reduce mechanical noise. To generate a sufficiently large dataset for machine learning, the servo motor was programmed to sweep the flex sensor from -90° to 90° (a full 180° range) in increments of 1°, repeating this cycle three times. This approach ensured a dense and uniformly distributed sampling of resistance values across the full bending range. Precise PWM signals controlled the servo sweep, simulating consistent and realistic bending motions across all cycles. In total, this procedure yielded approximately 540 data points per cycle, resulting in a high-resolution dataset well-suited for training and validating machine learning models. Overall, the hardware setup prioritized high repeatability, minimal noise, and ease of scalability for potential future applications.

- `analogRead(sensorPin)` measures the voltage across the flex sensor.
- Converts voltage reading into resistance using the voltage divider formula:

$$R_{\text{flex}} = R_{\text{fixed}} \times \left(\frac{V_{\text{in}} - V_{\text{out}}}{V_{\text{out}}} \right)$$

- Resistance is converted to kΩ for readability.
- Final output is a clean CSV-formatted line: Angle (degrees), Resistance (kΩ).

For the machine learning component, multiple algorithms were explored to model the relationship between sensor resistance and bending angle. Linear Regression served as the baseline to assess simple linear trends. Polynomial Regression (degree 4) was used to capture the flex sensor's inherent non-linear behavior. Tree-based methods like Random Forest and XGBoost were employed due to their ability to handle non-linearities and interaction effects efficiently. Gaussian Process Regression (GPR) was selected for its ability to provide not only high accuracy but also predictive uncertainty estimates, crucial for safety-critical applications. Finally, a Stacked Ensemble model combined the strengths of the aforementioned techniques to further improve predictive performance.

III. RESULTS

This section presents a comprehensive evaluation of the ML models trained to predict the bending angle of a flex sensor based on resistance measurements. Multiple models were tested to determine how well they capture the relationship between sensor resistance and angular displacement. Performance metrics include MAE, RMSE, MSE, R^2, Adjusted R^2, and 5-fold Cross-Validation scores.

To contextualize the effectiveness of the ML models, a baseline "no ML" approach was evaluated. This model simply predicted the mean bending angle for all input samples. It yielded an MAE of **43.72**°, RMSE of **51.26**°, MSE of **2627**, and an R^2 score of **-0.0002**. These results confirm that the baseline performs poorly, offering no real predictive power. All ML models outperformed this naive strategy by a substantial margin, validating the benefit of learning-based approaches for modeling flex sensor data.

Table 1 provides a summary of all key metrics for quick comparison. And Figure. 2 and 3 shows a sample of the plots used to extract the data in the table for R^2.

A. Linear Regression

The linear regression model achieved an MAE of **9.49**°, RMSE of **13.25**°, MSE of **175.69**, and R^2 score of **0.9331**, with an Adjusted R^2 of **0.9327**. These values indicate that a simple linear model is able to explain approximately 93% of the variance in the target bending angle. The slope of the fitted line was 50.40, with an intercept of 85.15, suggesting that as resistance increases, the predicted bending angle increases significantly.

While linear models offer interpretability, their primary limitation in this context is the assumption of linearity across the entire range of sensor responses. Given the non-linear nature of material strain and resistance, especially at higher angles, more flexible models may capture this behavior more effectively.

B. Polynomial Regression (Degree 4)

Polynomial regression of degree 4 improved upon the linear model, achieving an MAE of **9.48**°, RMSE of **13.02**°, MSE of **169.44**, R^2 of **0.9355**, and an Adjusted R^2 of **0.9351**. This suggests the model benefited from additional non-linear terms that better represent the physical behavior of the flex sensor.

In practical terms, the 4th-degree polynomial model captured bending curve nuances better than the linear approach, especially at extremes of the resistance range where linearity breaks down. The high Adjusted R^2 also suggests the additional terms were not superfluous but contributed meaningfully to model accuracy.

C. Random Forest Regression

The random forest model produced an MAE of **10.46**°, RMSE of **13.99**°, MSE of **195.79**, with an R^2 of **0.9255** and Adjusted R^2 of **0.9250**. Cross-validation yielded an average R^2 of **0.9391**, with CV MAE and RMSE of **9.47**° and **12.81**°, respectively.

Although not the most accurate model on the test set, the random forest performed well in cross-validation, suggesting good generalization. Its ability to model non-linear relationships without requiring extensive parameter tuning makes it a strong candidate for deployment in noisy environments or with sensor arrays.

D. XGBoost Regression

XGBoost, a gradient boosting framework, achieved an MAE of **10.09**°, RMSE of **13.63**°, MSE of **185.82**, R^2 of **0.9293**, and Adjusted R^2 of **0.9288**. Its cross-validation scores further support strong generalization: average R^2 was **0.9407**, CV MAE was **9.34**°, and CV RMSE was **12.65**°.

In the context of this study, XGBoost demonstrated resilience to overfitting and outlier resistance, valuable for real-time applications where sensor behavior may fluctuate. It serves as an effective trade-off between interpretability and predictive performance.

E. Gaussian Process Regression

Gaussian Process Regression yielded an MAE of **9.39**°, RMSE of **12.94**°, MSE of **167.53**, R^2 of **0.9362**, and Adjusted R^2 of **0.9358**. Unlike the previous models, GPR also provides a confidence interval for predictions, which could be useful in applications requiring uncertainty quantification.

This model captured complex, smooth relationships between resistance and bending angle and may be especially useful in safety-critical applications such as prosthetics or remote actuation systems where certainty is as important as accuracy.

F. Stacked Ensemble Model

The stacked model, which combined predictions from Random Forest, XGBoost, Polynomial Regression, and GPR, achieved an MAE of **9.69**°, RMSE of **13.23**°, MSE of **175.13**, and R^2 of **0.9333**, with an Adjusted R^2 of **0.9329**. Although this ensemble did not surpass the best individual model, its performance was consistent and robust across the test set.

The ensemble's results suggest that while combining multiple models may balance out individual weaknesses, it does not always guarantee superior performance. However, its robustness across different sensor readings indicates strong generalization potential.

Table 1: Performance comparison of regression models

Model	MAE (°)	RMSE (°)	MSE	R^2	Adjusted R^2
Baseline (No ML)	43.72	51.26	2627.45	−0.0002	
Linear Regression	9.49	13.25	175.69	0.9331	0.9327
Polynomial (Deg 4)	9.48	13.02	169.44	0.9355	0.9351
Random Forest	10.46	13.99	195.79	0.9255	0.9250
XGBoost	10.09	13.63	185.82	0.9293	0.9288
Gaussian Process	9.39	12.94	167.53	0.9362	0.9358
Stacked Model	9.69	13.23	175.14	0.9333	0.9329

Overall, machine learning models, particularly Gaussian Process Regression, substantially improved prediction accuracy compared to baseline methods.

IV. CONCLUSION

This study explored machine learning models to optimize bending angle prediction from flex sensor data. While Linear Regression provided a respectable baseline, its assumption of a linear relationship limited its accuracy, especially at higher angles where the sensor's response curve becomes nonlinear. Polynomial Regression (degree 4) better captured this curvature, improving predictive performance by modeling more complex bending behaviors. Among tree-based models, Random Forest handled non-linearities and feature interactions well but was ultimately outperformed by XGBoost. XGBoost's advantage stemmed from its gradient boosting mechanism, which sequentially corrects errors and reduces bias, leading to higher cross-validation scores and better generalization. GPR achieved

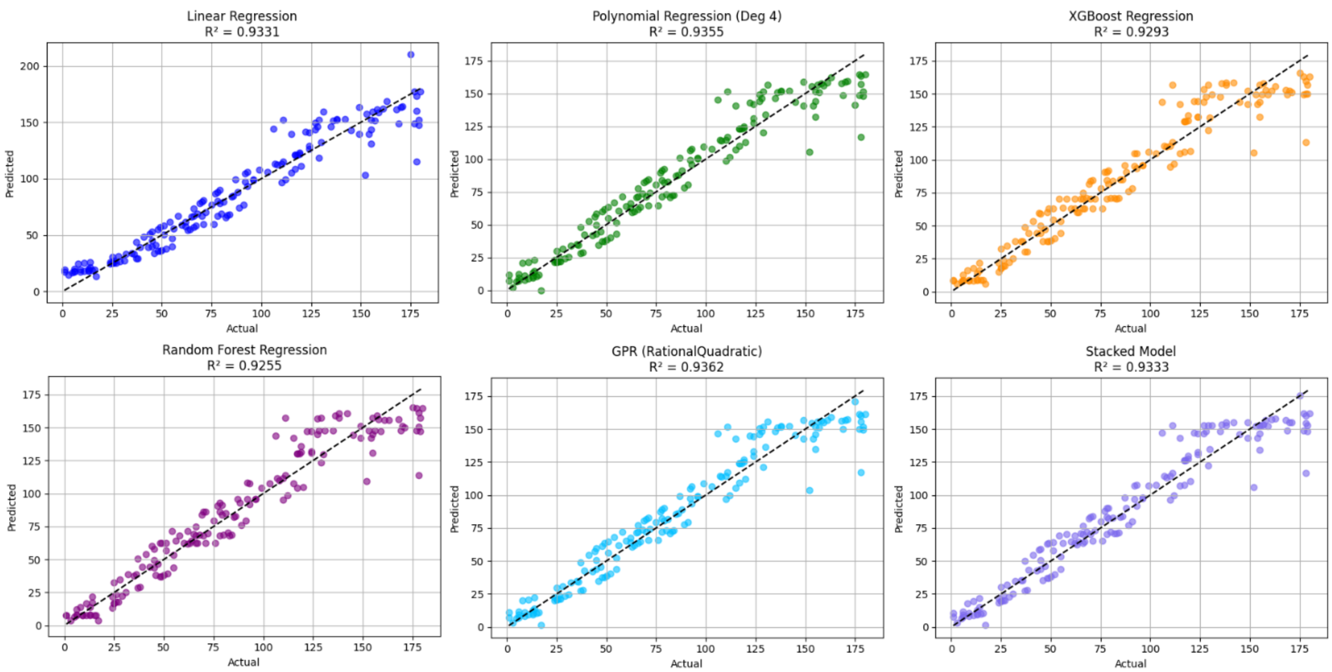


Fig. 2: R^2 comparison across models.

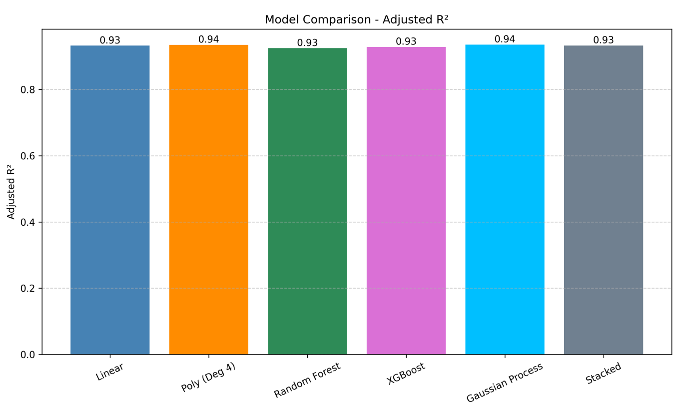


Fig. 3: Adjusted R^2 comparison across models.

the best overall performance, excelling not only in accuracy but also by providing uncertainty estimates—a valuable feature for safety-critical or real-time applications. The results validate that replacing static calibration tables with machine learning models enables more accurate, adaptive, and robust flex sensor systems, advancing their deployment in wearable technology, soft robotics, and gesture recognition.

Future work can extend this research by incorporating multi-sensor fusion with modalities such as IMUs, pressure sensors, or EMG to improve system robustness and gesture recognition. Temporal modeling with LSTM, GRU, or Transformer networks could enable dynamic, real-time tracking of movements. To support wearable deployment, lightweight and quantized models could be optimized for micro-controllers like the Arduino Nano 33 BLE. Adaptive learning strategies that personalize predictions based on user data would minimize retraining needs over time. Additionally, implementing advanced filtering or recurrent models could enhance noise resilience and correct for sensor drift. Finally, validating model performance across different temperatures, humidity levels, and extended use cycles would ensure long-term stability and reliability in practical applications.

ACKNOWLEDGMENT

This work is supported by National Science Foundation Grant No. 2431589. We are thankful to Wally Lozano Diaz for implementing the bending device.

REFERENCES

[1] U.S. Census Bureau, "Disability in america 2023." https://www.census.gov/, 2023. Accessed: 2025-03-01.

[2] Centers for Disease Control and Prevention, "Disability impacts all of us infographic," n.d. Accessed: 2025-04-09.

[3] R. Tchantchane, H. Zhou, S. Zhang, and G. Alici, "A review of hand gesture recognition systems based on noninvasive wearable sensors," *Advanced intelligent systems*, vol. 5, no. 10, p. 2300207, 2023.

[4] C. C. S. Soares, L. A. Silva, A. Fernandes, G. V. González, V. R. Leithardt, and W. D. Parreira, "Intelligent sensors in assistive systems for deaf people: a comprehensive review," *PeerJ Computer Science*, vol. 10, p. e2411, 2024.

[5] F. Salman, Y. Cui, Z. Imran, F. Liu, L. Wang, and W. Wu, "A wireless-controlled 3d printed robotic hand motion system with flex force sensors," *Sensors and Actuators A: Physical*, vol. 309, p. 112004, 2020.

[6] T. M. N. U. Akhund, Z. A. Shaikh, I. D. L. T. Díez, M. Gafar, D. H. Ajabani, O. Alfarraj, A. Tolba, H. Fabian-Gongora, and L. A. D. López, "Iost-enabled robotic arm control and abnormality prediction using minimal flex sensors and gaussian mixture models," *IEEE Access*, vol. 12, pp. 45265–45278, 2024.

[7] A. K. Panda, R. Chakravarty, and S. Moulik, "Hand gesture recognition using flex sensor and machine learning algorithms," in *2020 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES)*, pp. 449–453, IEEE, 2021.

[8] D. Feng, C. Zhou, J. Huang, G. Luo, and X. Wu, "Design and implementation of gesture recognition system based on flex sensors," *IEEE Sensors Journal*, vol. 23, no. 24, pp. 31389–31398, 2023.

[9] H.-G. Doan and N.-T. Nguyen, "Fusion machine learning strategies for multi-modal sensor-based hand gesture recognition," *Engineering, Technology & Applied Science Research*, vol. 12, no. 3, pp. 8628–8633, 2022.

[10] SparkFun Electronics, "Flex sensor hookup guide: Flex sensor overview," n.d. Accessed: 2025-04-29.