Smartphone-based Portable Blood Parameter Sensing using Machine Learning

Abstract—**In this work, a smartphone-based miniaturized platform for blood sensing has been developed. Biochemical blood assays develop colorimetric changes depending on the parameter present in the blood. The developed sensor works on the principle of colorimetric. The image of the sample after mixing with the reagent is captured and analyzed using a smartphone. Machine learning is used to predict the blood parameter concentration using the extracted information from the image. An android application is developed which analyzes the captured image and estimates the output concentration. The developed platform is capable of quantizing all biochemistry performed in the visible range. To verify the functioning of the developed Android application, the platform was tested with Uric acid, Creatinine, Albumin, Glucose, and Calcium. These tests are required to analyze renal functioning. The results are compared with the commercial analyzer. The developed platform is fully integrated, portable, easy to use, compact, economical, and well-suited for home testing applications.**

Keywords—Smartphone, biochemistry, colorimetric, Uric acid, renal functioning

I. Introduction

The biochemistry analyzer is medical equipment that is frequently used in clinical labs for albumin testing, sugar level tests, or to find levels of enzymes and creatinine in the blood. Healthcare professionals and medical practitioners frequently employ biochemistry analysers in a variety of medical settings and laboratories to diagnose, treat, and detect serious conditions, such as health disorders [1]. The biochemistry analyzer works on the principle of Beer-Lambert law and requires complex optical assembly [2]. In recent years much research has been done for biochemical blood sensing using methods like electrochemical [3][4], Fluorescence [5], optical detection[6][17], Colorimetric[7][8], and smartphone-based analyzer [9][10][11][15].

The biochemistry analyzer is the gold standard used in pathological laboratories and is based on the colorimetric detection principle. The basic principle of detection lies around the fact that all the biochemistry in the visible range will develop the color depending upon the concentration of the targeted analyte present in the blood sample. The biochemical analysis uses centrifuged blood i.e., blood serum which is transparent in color. For each biochemistry a reagent kit is available, when mixed with the serum changes the color of the serum depending upon the concentration present for measuring blood parameters. In the existing analyzer, this color change is quantized, by measuring the absorbance, which requires a costly mechanical filter. But our idea is to measure this color change more competently and easily i.e., using color image analysis. Also, the conventional biochemistry analyzer is mostly expensive, has complex instrumentation, is time-consuming, and demands continuous maintenance due to huge hardware dependency.

Hence, there is a need to develop an instrument for the biochemical analysis of blood capable of overcoming all the mentioned drawbacks. Hence, for the visible-range-based analysis, we have developed a platform with a colorimetric detection unit based on color image acquisition and analysis. From the color image of the test sample, absorbance is calculated and used for analysis. This calculation of absorbance from color images using a smartphone removes the requirement for wavelength filters and minimizes the hardware [12]. The blood tests namely Uric acid, Creatinine, Calcium, Albumin, and Glucose are used to identify renal functioning-related diseases.

To achieve this, we have used the red (r), green (g), and blue (b) values extracted from images, and employed a diverse array of machine learning and deep learning models, each tailored to the unique characteristics of the data at hand. The training process involves fine-tuning model parameters to optimize performance and generalization. The comparative analysis of these models forms a crucial component of our investigation, shedding light on the most effective approaches for accurate uric acid level prediction. Hence, the detailed process for uric acid is explained in this work and the experimental results of the remaining parameters are summarized in this paper. Machine learning in healthcare must be prioritized to get better results[16]. The blood uric-acid level prediction were taken into consideration for this paper work.

II. Experimental

A. Chemicals used

A uric acid reagent is used for sensing uric acid concentration in blood. The uric acid reagent consists of Phosphate Buffer (85mmol/L), Uricase (1.6KU), Peroxidase (1.75KU), 4-Amino Antipyrine (0.55mmol/L), Chromogen, Surfactant, and Stabilizers and standard Concentration is 8mg/dl. For creatinine, sensing modified Jaffe's method is used and the reagent consists of picric acid (20mmol/L). sodium hydroxide (15mmol/L). For albumin reagent consists of succinate buffer (90mmol/L) and bromocresol green (0.26 mmol/L). Glucose reagent consists of glucose oxidase (13KU/L) and peroxidase (2 KU/L). Calcium reagent consists of hydroxyquinoline (100mmol/L), arsenazo III (200 mmol/L).

B. Developed Blood Parameter Sensing platform

To address the drawbacks of existing analyzers, in this work, a fully integrated, portable, inexpensive, machine learning-driven smartphone-based biochemistry analyzer is developed. The experimental setup for biochemical blood sensing is shown in Fig. 1. The crucial role is to capture the image of the test sample flawlessly. For this purpose, a robust enclosure is designed, it is a 3D-printed black box to avoid any interference of light. A LED strip with a diffuser is mounted on the top surface to have sufficient illumination required for

capturing images. Any Android-based smartphone can be used to capture the image of the test sample.

Initially after the sample preparation, the sample is kept in a flow cell and placed in the detection unit. The smartphone captures the image of the test sample present in the flow cell. The captured image is cropped and averaged to filter out any random noise. From the cropped ROI, red (r), green (g), and blue (b) pixel values are extracted. The absorbance is calculated using the following equation [7].

$$Absorbance\ (A) \propto -log10\ (r + g + b) \quad (1)$$

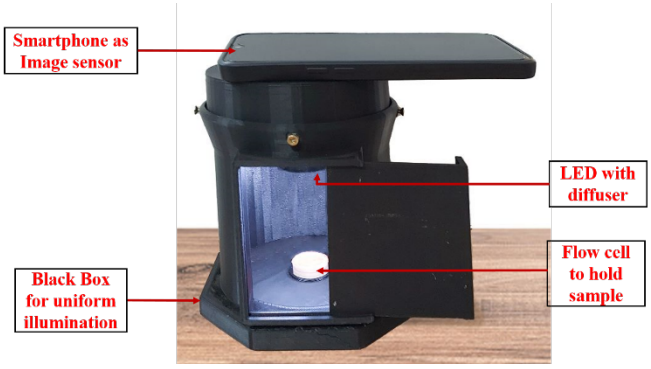

Fig. 1. Experimental setup

C. Architectural Diagram

Data preparation is essential when you get your dataset. After mixing the solution with the reagent, we collected the R, G, and B values for data processing after leaving the solution for 15 minutes. This is important since accurate analysis and dependable study results depend on it.

Later, we used gradient boosting, Support Vector Regression (SVR), Artificial Neural Networks (ANN), and linear regression to forecast the concentrations of uric acid in solutions. Using a variety of methods improves resilience and accuracy, offering a complete solution for trustworthy forecasts that are essential to our research projects and analytical procedures. Fig 1.1 depicts the architectural diagram we implemented in this study.

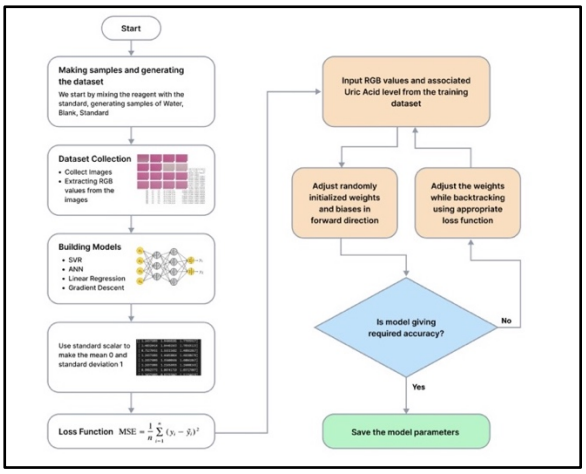

Fig 1.1 Architectural Diagram

We evaluate different machine learning models using Mean Squared Error (MSE) as our selected loss function. This metric enables thorough assessment and guarantees accuracy when quantifying the difference between expected and actual values, which is crucial for fine-tuning and optimizing model results inside our analytical framework.

III. Results and Discussion

In the commercial biochemistry analyzer, the colorimetric change is observed by calculating the absorbance of light using the Beer-Lambert law. We propose to measure absorbance from color image analysis. A smartphone is used to capture images in the color space. Red, Green, and Blue are the fundamental colors from which all other colors are created. Red, green, and blue filters on the camera sensor are employed to divide the input light into its three primary hues. The RGB value of that specific pixel is then used to store the intensity of each color. Such pixels are combined to create an image. These RGB values can be used to extract a wealth of data from the image. In our case, we are measuring absorbance from these RGB values.

A. Estimation of Uric acid concentration

The result of purine metabolism in humans is uric acid. To diagnose and treat a variety of renal and metabolic problems, such as renal failure, gout, leukemia, psoriasis, and malnutrition, as well as in patients taking cytotoxic medications, the blood's uric acid concentration is measured. The abnormal ability to process the byproducts of regular cellular breakdown or excessive cellular destruction are the two factors that connect these illnesses (gout). Uric acid is oxidized by uricase in this technique. Peroxidase then reacts with the peroxide created by this reaction in the presence of 4 aminophenazone to create a quantifiable colored product.

$$\text{Uric Acid} + O_2 + H_2O \xrightarrow{\text{Uricase}} \text{Allantoin} + CO_2 + H_2O_2 \quad (2)$$

$$\text{DHBS} + 4\text{AAP} + 2H_2O_2 \xrightarrow{\text{Peroxidase}} \text{Quinoneimine} + 4H_2O \quad (3)$$

The absorbance of this Quinoneimine dye is proportional to uric acid concentration in the sample.The absorbance of the sample (As) is calculated with reference to the reagent blank as follows.

$$(As) \propto -\log 10 \frac{rs + gs + bs}{rb+gb+bb} \quad (4)$$

Where, the rs, gs, and bs are the RGB values of the sample, and rb, gb, and bb are the RGB values of the blank. Initially, the RGB values have to be normalized by dividing RGB values by the standard deviation.

$$R, G, B = r, g, b / \sqrt{r^2 + g^2 + b^2} \quad (5)$$

Hence, the absorbance is modified as,

$$(As) \propto -\log 10 \frac{Rs + Gs + Bs}{Rb+Gb+Bb} \quad (6)$$

The camera sensitivity needs to be taken into consideration, to get accurate analysis as shown in Fig.2. We pass a particular wavelength of the light on the camera sensor to get RGB values.

.

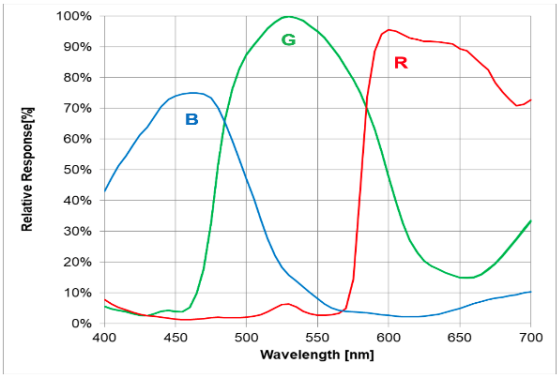

Fig. 2. Sensitivity calculation of R, G, and B.

These values when normalized can be used as the sensitivity of the camera for that wavelength. $Sr = rw/255$, $Sg = gw/255$, $Sb = bw/255$. Where, Sr, Sg, and Sb are the sensitivity of red, green, and blue pixels of the camera at a given wavelength of light, and, rw, gw, and bw are the actual 8-bit RGB values measured by the camera at the given wavelength. Hence, the absorbance becomes,

$$As \propto -log10 \frac{Sr*Rs+Sg*Gs+Sb*Bs}{Sr*Rb+Sg*Gb+Sb*Bb} \quad (7)$$

$$As = -K\ log10 \frac{Sr*Rs+Sg*Gs+Sb*Bs}{Sr*Rb+Sg*Gb+Sb*Bb} \quad (8)$$

Where K is the proportionality constant. The absorbance of a test sample is the difference between measured absorbance from the blank absorbance.

$$At = Asample - Ablank \quad (9)$$

Using Beer-Lambert law, considering uniform molar coefficients and path length, the concentration of the unknown sample is calculated using (10) [6],

$$Ct = Cs * (At\ /As) \quad (10)$$

Where Ct is the concentration of the test sample to be measured, Cs is the concentration of the uric acid standard (8 mg/dL), As is the absorbance of the standard, At is the absorbance of a test sample. Hence, any unknown concentration is calculated using (8).

B. Machine learning

Uric acid known samples are prepared for the concentration ranging from 1 mg/dL to 20 mg/dL with the step of 0.1 mg/dL using the uric acid standard. All prepared samples are tested, and RGB values are recorded every 3 seconds for 2 minutes and plotted as shown in Fig. 3. So we got a total of 120 RGB value pairs for each prepared sample.

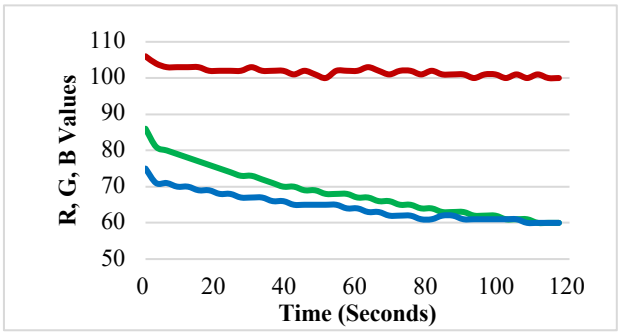

Fig. 3. R, G, and B values for the initial 120 seconds of uric acid sample.

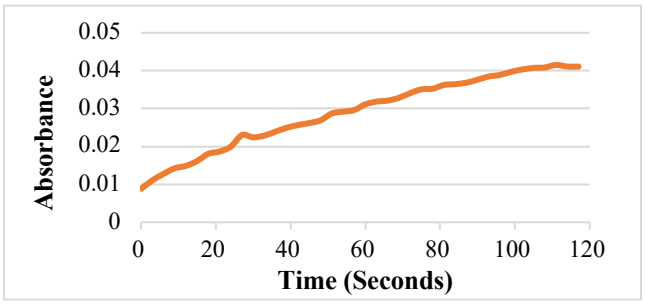

Fig. 4. Absorbance plot from R, G and B values

The obtained RGB values are further used to obtain absorbance with respect to time as shown in Fig. 4. The machine learning logarithmic regression model is applied on this graph for the further prediction of absorbance at 10th minute (600 seconds). The accuracy achieved was around 95.25%.

C. Linear Regression

Linear regression is a statistical technique for modelling the connection between a dependent variable and one or more independent variables. Finding the best-fitting straight line, or hyperplane in the case of several independent variables, that minimizes the sum of squared differences between the predicted and actual values is the main objective of linear regression.

Generally speaking, the equation for a basic linear regression model with one independent variable is as follows:

$Y = \beta_0+\beta_1X+F$

In this case, the dependent variable is Y which in our case is the uric-acid levels.

This variable X is independent which in our case are the R,G,B values.

Linear Regression was also used to forecast island submergence as a result of the combined effects of sea level rise and glacier melting [13]. Thus, linear regression proved its capacity in predicting real valued data.

The best fit line for the linear regression is depicted in the fig 5.1, showing how well it matches the data points. It also shows error bars, which are crucial for assessing the accuracy of the model since they show the difference between the actual data and the anticipated values on the line

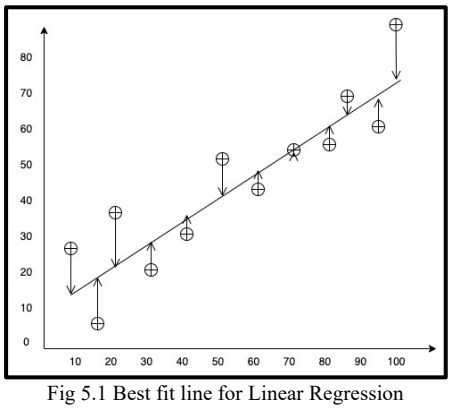

Fig 5.1 Best fit line for Linear Regression

In this work we have used linear regression models in our study to draw conclusions from our dataset, which comprises inputs for RGB color values and outputs for

concentration. We may investigate relationships and forecast concentration levels based on color features using the linear regression model. Fig 5.2 represents scatterplot with Regression line, this plot compares actual concentration values to predictions made by our linear regression model for concentration values. The actual and anticipated values would line up in a perfect match, which is represented by the red dashed line.

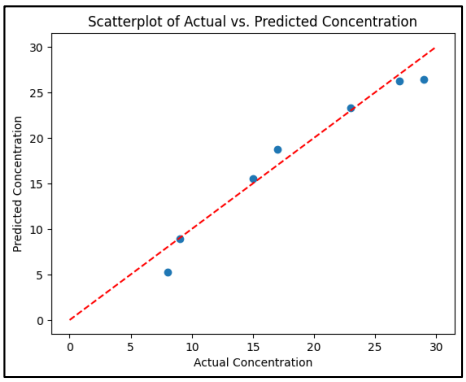

Fig 5.2 Scatter plot with Regression Line

Below are the results of linear regression model:

Mean Square Error: 2.6475176280426522
R-Squared: 0.9556028871409685
Coefficients: [[-0.31562943 ,0.13938635, -0.83398104]]
Intercepts: 188.78030574
The equation thus generated is: -0.31562943*(r)+ 0.13938635*(g)+ -0.83398104*(b)+ 188.78030574

D. Gradient Boosting

A potent machine learning method for both regression and classification applications is the gradient boosting. It is a member of the ensemble learning family and excels at constructing predictive models by fusing the results of several subpar models—most frequently, decision trees—to produce a more robust and precise model. Gradient Boosting relies on iteratively addressing the shortcomings of prior models; therefore, the word "boosting."

A key optimization algorithm in machine learning, gradient boosting, is graphically depicted in the above figure fig 5.3. It's a fundamental idea in machine learning model training since it shows how a parameter's value iteratively changes in the direction of steepest descent to minimize a cost function.

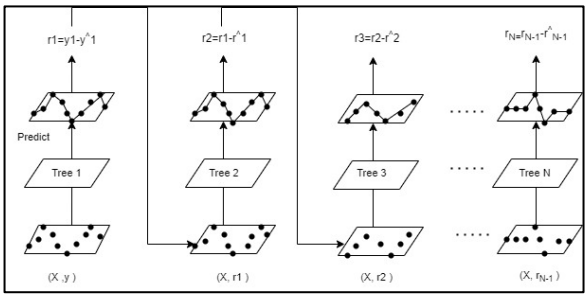

Fig 5.3 Gradient Boosting

Gradient Boosting achieves state-of-the-art performance through ensemble learning, yielding robust and accurate outcomes in a variety of machine learning problems. It performs better than other models in regression, classification, and ranking because it can successfully handle a variety of datasets, reduce overfitting, and identify intricate patterns in the data. Fig 5.4 and Fig 5.5 represents actual and predicted data points and Fig 5.6 depicts a combined plot of actual and predicted points.

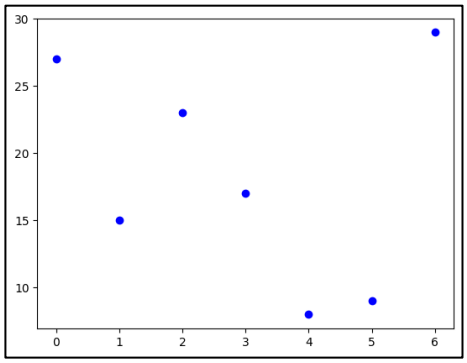

Fig 5.4 Actual Data Points

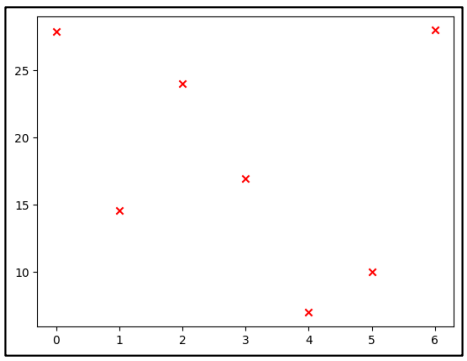

Fig 5.5 The Data Points predicted by our model

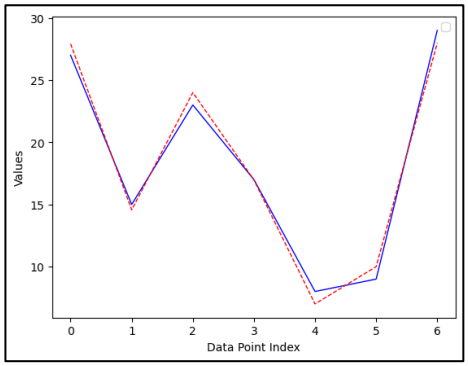

Fig 5.6 Combined Graph

Below are the results for the gradient boosting,

Mean Squared Error: 0.7221651174693876
R-Squared: 0.9878897704462697

E. Artificial Neural Networks

A machine learning model called an Artificial Neural Network (ANN) is modelled after the neural architecture of the human brain. It is made up of layers of connected nodes, or artificial neurons. An input layer, one or more hidden layers, and an output layer are the most common types of

these layers. ANNs are utilized in many different applications, including predictive modelling, natural language processing, and image identification.

Structure: An input layer, one or more hidden layers, and an output layer make up the structure of ANNs. Every neuron in one layer of the network is connected to every other layer's neuron, allowing information to flow through the network. In order to catch intricate patterns in the data, the associated weights of the connections are modified during training. Artificial Neural Network was earlier used to predict the weather accurately [18], to categorize earthquake vibrations [19] and also for processing fingerprint image noise [20].

An Artificial Neural Network (ANN) model's architecture, comprising its input layer, hidden layers, and output layer, is shown in detail in the fig 5.7.

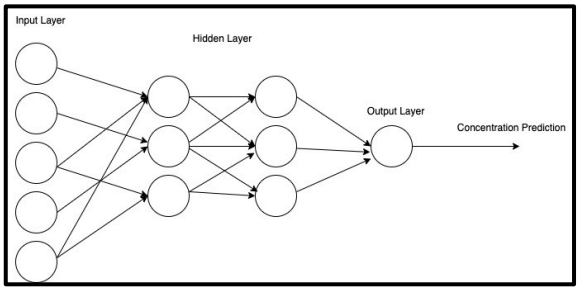

Fig 5.7 Artificial Neural Network's structure

The size of the dataset that ANNs require is one of its limitations. Big, varied datasets are ideal for training ANNs, and this is where they flourish. In cases of overfitting, when the network performs well on training data but poorly on unknown data, small datasets may be the cause. To overcome this constraint, methods such as transfer learning and data augmentation are used to maximize the available data. We trained the dataset for 2000 epochs and the loss function was 'Mean Squared error' also using 'Adam' as an optimizer. Fig 5.8 represents loss vs epochs curve and Fig 5.9 represents true vs predicted value graph for ANN.

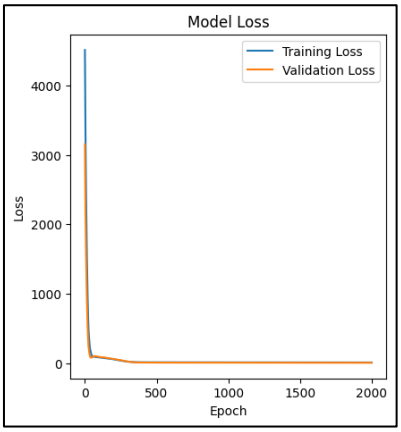

Fig 5.8 Loss vs Epochs

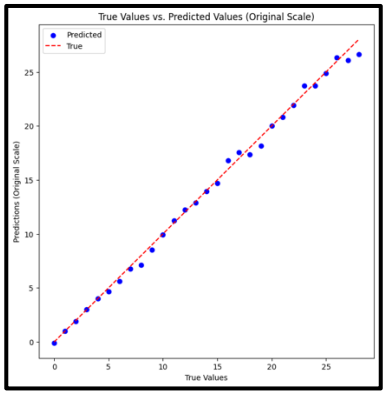

Fig 5.9 True vs Predicted values for ANN

Mean Squared Error (MSE): 0.24124931785320233
R-squared: 0.9965535811735257
Final error percentage for uric-acid level prediction with artificial neural network: 2.7431619300199634%

F. Support Vector Regression

Overview: Support Vector Regression (SVR) seeks to identify a hyperplane that maximizes the epsilon-tube error margin while fitting the training data. The goal is to keep this margin as small as feasible while making sure that the majority of the data points fall inside it. Data points that sit outside or near the margin are of particular interest in SVR because they have a big impact on the regression model.

Support Vector Regression (SVR) is illustrated in fig 5.10, where the focus is on the support vectors—that is, the important data points that have a major impact on the decision boundary of the model. An additional figure fig 5.11 demonstrates how data is transformed via the kernel function of the SVR, clarifying how non-linear mapping improves the model's performance.

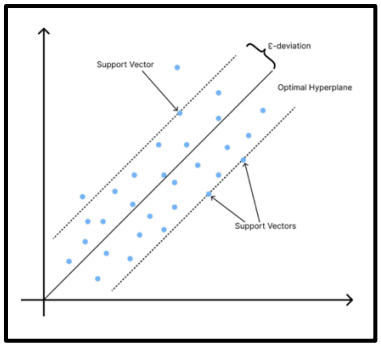

Fig 5.10 Decision boundary

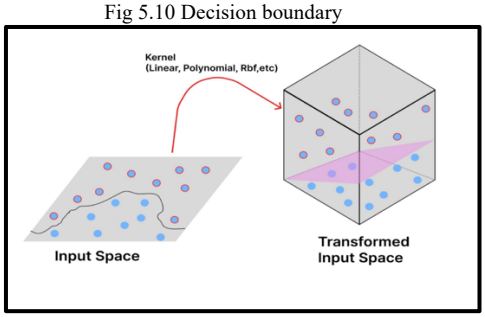

Fig 5.11 Transformed Input space and Kernel Function

A comparative analysis of SVR and linear regression was demonstrated on Shale gas in year 2016 [14], which illustrated the capabilities of SVR.

Regression tasks are handled using the machine learning method Support Vector Regression (SVR). It is an improvement over Support Vector Machines (SVM) and works especially well with tiny datasets. SVR's resilience and generalizability are its main advantages over Artificial Neural Networks (ANN) for small datasets. Finding the hyperplane that minimises the margin of error and best matches the data is the main goal of SVR. Because of this margin control, SVR is less likely to overfit, which is a common problem with ANNs when working with small amounts of data. The following fig 5.12 represents the predicted versus the actual value for Support Vector Regression (SVR).

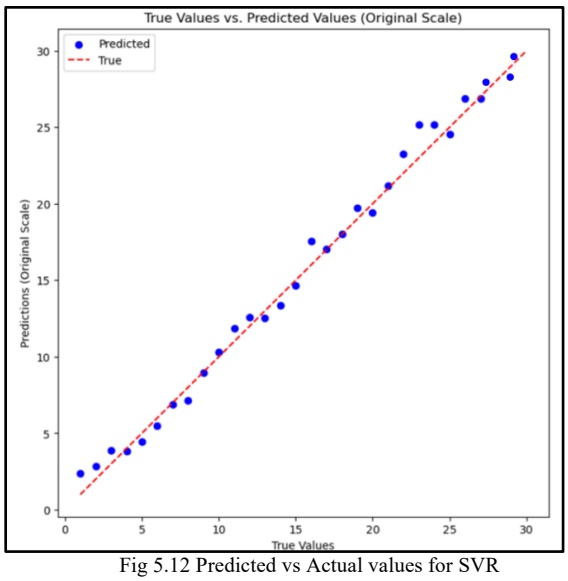

Fig 5.12 Predicted vs Actual values for SVR

Mean Squared Error: 1.00665147035270746
R Squared Error: 0.9936571387549996
Final Error Percentage: 4.249046665809653 %

K-fold cross-validation is a widely used method for assessing the effectiveness and generalizability of the SVR model. Since K is set to 4 in this instance, the dataset is split up into four smaller groups. Three subsets are used to train the model, and the remaining subset is used for testing. In order to guarantee that every subset acts as a test set precisely once, this procedure is performed for every convolution. Compared to just dividing training and testing, K-fold cross-validation offers a more thorough evaluation of the model's performance and aids in locating issues with over- or under-fitting.

Metrics like the coefficient of determination (R-squared), mean squared error (MSE), and mean absolute error (MAE) are typically used to assess the performance of SVR models. An impressive performance was shown by the model, which explained 93.18% of the variance in uric acid levels in this investigation, with an R-squared (R2) of 0.9318. When the SVR model's R2 value is near to 1, it means that the biochemistry analyzer's RGB values can be used to make predictions that the model matches the data well.

Artificial Neural Network (ANN) Support Vector Regression (SVR) in prediction using levels from RGB data, with a reduced error rate (4% vs. 2%). Because ANN performs better than SVR, it may be more appropriate for this task because of its capacity to handle complex relationships in the data.

TABLE I. PERFORMANCE ANALYSIS

Machine Learning Model	MSE	R squared
Linear Regression	2.6475176280426522	0.9556028871409685
Gradient Boosting	0.7221651174693876	0.9878897704462697
Support Vector Regression	1.00665147035270746	0.9936571387549996
Artificial Neural Networks	0.24124931785320233	0.9965535811735257

TABLE II. ERROR PERCENTAGE

Machine Learning Model	Percentage
Artificial Neural Network	2.7431619300199634
Support Vector Regression	4.249046665809653

G. *Repeatability analysis*

The degree of consistency between the results of subsequent tests of the same sample taken under different measurement conditions is known as repeatability. For intra-repeatability, the same sample was tested five times at the same conditions, and for inter-repeatability, the sample with the same concentration was prepared five different times and tested in different conditions. The repeatability analysis of the developed system is summarized in Table III.

TABLE III. INTER & INTRA REPEATABILITY ANALYSIS

Control serum	Conc.	Mean	SD	%CV
Intra- repeatability (n=5)				
Sample 1	50	259.25	0.089	0.035
Sample 2	100	507.31	0.019	0.004
Sample 3	150	754.69	0.004	0.001
Sample 4	200	1009.45	0.045	0.004
Sample 5	250	1254.06	0.345	0.028
Inter-repeatability (n=5)				
Sample 1	50	282.22	0.14	0.049
Sample 2	100	508.27	0.60	0.117
Sample 3	150	762.37	0.96	0.125
Sample 4	200	1010.4	1.61	0.160
Sample 5	250	1260.85	1.36	0.107

H. *Result comparison with the commercial analyzer*

To verify the functioning of the developed biochemical blood-sensing platform, the results are compared with the commercial biochemistry analyzer. From below Table II, it is evident that the developed sensing platform is suitable to be used in practical applications.

TABLE IV. RESULT VALIDATION WITH COMMERCIAL ANALYZER

Test	Analyzer	Results (mg/dL)				
Uric acid	Commercial	278	123	158	101	172
	Developed	262	118	149	97	167
Creatinine	Commercial	1.23	1.41	2.52	3.18	3.56
	Developed	1.20	1.37	2.48	3.11	3.41

Albumin	Commercial	5.2	8	2	4.2	6.1
	Developed	5	7.1	1.8	4	5.9
Glucose	Commercial	278	123	158	101	172
	Developed	262	118	149	97	167
Calcium	Commercial	2.5	3.44	5.9	8.77	14.2
	Developed	2.6	3.57	6.12	8.84	14.2

I. BioChem: Android application

Android Studio is used to develop the Android application BioChem for biochemical blood sensing. The screenshots are shown in Fig. 5.

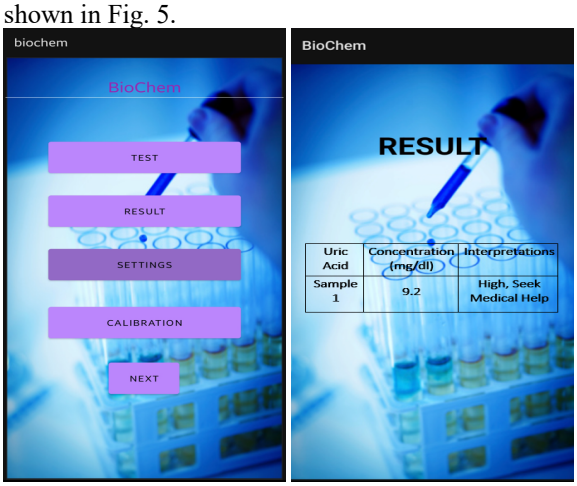

Fig. 4. Android application screenshots

IV. Conclusion

In this work, a portable biochemical blood-sensing platform has been developed. The developed platform works on the principle of colorimetry. The sensing unit consists of a smartphone and a robust compact environment for image capturing and analysis. An Android application is developed to perform all the analyses. The developed sensing platform is tested for Uric Acid, Sodium, Albumin, Cholesterol Glucose, Total Protein, Albumin, Triglyceride, Bilirubin, etc. and the results are matching with the commercial analyzer. These tests with a portable platform help the patient to monitor the renal functioning at home. Various machine learning algorithms were fine-tuned and their results were compared to get the best suitable algorithm for this work. Support vector regression outperformed other algorithms and also showcased its capacity to handle complex relationships in data.

Acknowledgment

The authors are grateful to the Centre for Microsystem, RCOEM Nagpur, for providing the research facilities. Financial support for this project has been provided through RCOEM YFRF Scheme.

References

[1] S. R. Taneja, R. C. Gupta, J. Kumar, K. K. Thariyan, and S. Verma, “Design and Development of Microcontroller-Based Clinical Chemistry Analyser for Measurement of Various Blood Biochemistry Parameters,” 2005.

[2] S. Kapoor and V. Goel, “Design and Interfacing of the Optical Assembly for Automated Analyzer,” Int. J. Sci. Res., vol. 2, no. 5, pp. 2319–7064, 2013, [Online]. Available: www.ijsr.net.

[3] C. Fang, H. Li, J. Yan, H. Guo, and T. Yifeng, “Progress of the Electrochemiluminescence Biosensing Strategy for Clinical Diagnosis with Luminol as the Sensing Probe,” ChemElectroChem, vol. 4, no. 7, pp. 1587–1593, 2017, doi: 10.1002/celc.201700465.

[4] M. Bhaiyya, P. K. Pattnaik, and S. Goel, “Portable Electrochemiluminescence Platform with Laser-Induced Graphene-Based U-Shaped Bipolar Electrode for Selective Sensing of Various Analytes,” IEEE Trans. Electron Devices, vol. 68, no. 5, pp. 2447–2454, May 2021, doi: 10.1109/TED.2021.3066083.

[5] S. Choi, E. Y. Choi, D. J. Kim, J. H. Kim, T. S. Kim, and S. W. Oh, “A rapid, simple measurement of human albumin in whole blood using a fluorescence immunoassay (I),” Clin. Chim. Acta, vol. 339, no. 1–2, pp. 147–156, 2004, doi: 10.1016/j.cccn.2003.10.002.

[6] S. D. Palekar and J. Kalambe, “Development of an Optical Detection-Based Universal Biochemical Blood Analysis Platform,” IEEE Sens. J., vol. 21, no. 20, pp. 22434–22441, 2021, doi: 10.1109/JSEN.2021.3108560.

[7] M. Ş. Niculescu, A. Florescu, and S. Paşca, “Automated portable biochemistry analyzer based on image acquisition,” Rev. Roum. des Sci. Tech. Ser. Electrotech. Energ., vol. 65, no. 3–4, pp. 271–276, 2020.

[8] S. Palekar, J. Kalambe, and R. M. Patrikar, “Biochemical Blood Sensing platform with CMOS Image Sensor and Software-based Wavelength Filter,” IEEE Sens. J., Nov. 2022, doi: 10.1109/JSEN.2022.3208810.

[9] Y. Fu and J. Guo, “Blood Cholesterol Monitoring with Smartphone as Miniaturized Electrochemical Analyzer for Cardiovascular Disease Prevention,” IEEE Trans. Biomed. Circuits Syst., vol. 12, no. 4, pp. 784–790, 2018, doi: 10.1109/TBCAS.2018.2845856.

[10] D. Xu, X. Huang, J. Guo, and X. Ma, “Automatic smartphone-based microfluidic biosensor system at the point of care,” Biosens. Bioelectron., vol. 110, pp. 78–88, 2018, doi: 10.1016/j.bios.2018.03.018.

[11] D. Zhang and Q. Liu, “Biosensors and bioelectronics on smartphone for portable biochemical detection,” Biosens. Bioelectron., vol. 75, pp. 273–284, 2016, doi: 10.1016/j.bios.2015.08.037.

[12] S. Palekar, J. Kalambe, and R. M. Patrikar, “Biochemical Blood Sensing platform with CMOS Image Sensor and Software-based Wavelength Filter,” IEEE Sens. J., 2022, doi: 10.1109/JSEN.2022.3208810.

[13] M. Huang, "Theory and Implementation of linear regression," 2020 International Conference on Computer Vision, Image and Deep Learning (CVIDL), Chongqing, China, 2020, pp. 210-217, doi: 10.1109/CVIDL51233.2020.00-99.

[14] Kavitha S, Varuna S and Ramya R, "A comparative analysis on linear regression and support vector regression," 2016 Online International Conference on Green Engineering and Technologies (IC-GET), Coimbatore, 2016, pp. 1-5, doi: 10.1109/GET.2016.7916627.

[15] Ö. B. Mercan and V. Kılıç, "Deep Learning based Colorimetric Classification of Glucose with Au-Ag nanoparticles using Smartphone," 2020 Medical Technologies Congress (TIPTEKNO), Antalya, Turkey, 2020, pp. 1-4, doi: 10.1109/TIPTEKNO50054.2020.9299296.

[16] K. Shailaja, B. Seetharamulu and M. A. Jabbar, "Machine Learning in Healthcare: A Review," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2018, pp. 910-914, doi: 10.1109/ICECA.2018.8474918.

[17] R. Bunescu, N. Struble, C. Marling, J. Shubrook and F. Schwartz, "Blood Glucose Level Prediction Using Physiological Models and Support Vector Regression," 2013 12th International Conference on Machine Learning and Applications, Miami, FL, USA, 2013, pp. 135-140, doi: 10.1109/ICMLA.2013.30.

[18] D. V. Rayudu and J. F. Roseline, "Accurate Weather Forecasting for Rainfall Prediction Using Artificial Neural Network Compared with Deep Learning Neural Network," *2023 International Conference on Artificial Intelligence and Knowledge Discovery in Concurrent Engineering (ICECONF)*, Chennai, India, 2023, pp. 1-6, doi: 10.1109/ICECONF57129.2023.10084252.

[19] F. A. Tasa, Istiqomah, M. A. Murti and I. Alinursafa, "Classification of Earthquake Vibrations Using the ANN (Artificial Neural Network) Algorithm," 2022 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT), BALI, Indonesia, 2022, pp. 102-107, doi: 10.1109/IAICT55358.2022.9887421.

[20] K. Han, "Artificial Neural Network for Processing Fingerprint Image Noise," 2022 23rd ACIS International Summer Virtual Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD-Summer), Kyoto City, Japan, 2022, pp. 9-14, doi: 10.1109/SNPD-Summer57817.2022.00011.