JURNAL INFOTEL
VOL. 17, NO. 2, MAY 2025, PP. 229–251. DOI:10.20895/INFOTEL.V17I2.1331

RESEARCH ARTICLE

Autism Face Detection System using Single Shot Detector and ResNet50

Melinda Melinda[1,*], Muhammad Fauzan Alfariz[2], Yunidar Yunidar[3], Agung Hilm Ghimri[4], Maulisa Oktiana[5], Rizka Miftahujjannah[6], and Nurlida Basir[7]

[1,2,3,4,5,6]Department of Electrical Engineering and Computer, Universitas Syiah Kuala, Banda Aceh 23111, Indonesia
[7]Faculty of Sains and Techology, Universiti Sains Islam Malaysia, Negeri Sembilan 71800, Malaysia

*Corresponding email: melinda@usk.ac.id

Received: February 25, 2025; Revised: March 18, 2025; Accepted: April 16, 2025.

Abstract: The facial features of children can provide important visual cues for the early detection of autism spectrum disorder (ASD). This research focuses on developing an image-based detection system to identify children with ASD. The main problem addressed is the lack of practical methods to assist healthcare professionals in the early identification of ASD through facial visual characteristics. This study aims to design a prototype facial image acquisition and detection system for children with ASD using Raspberry Pi and a deep learning-based single-shot detector (SSD) algorithm. In this method, the face detection model uses a modified ResNet50 architecture, which can be used for advanced analysis for classification between autistic and normal children, achieving 95% recognition accuracy on a dataset consisting of facial images of children with and without ASD. The system is able to recognize the visual characteristics of the faces of children with ASD and consistently distinguish them from those of normal children. Real-time testing shows a detection accuracy ranging from 86% to 90%, with an average accuracy of 90%, despite fluctuations caused by variations in movement and viewing angle. These results show that the developed system offers high accuracy and has the potential to function as a reliable diagnostic tool for the early detection of ASD, which ultimately facilitates timely intervention by healthcare professionals to support the optimal development of children with ASD.

Keywords: autism spectrum disorder, deep learning, image acquisition, raspberry pi, single shot detector

1 Introduction

Autism Spectrum Disorder (ASD) is a developmental condition that affects an individual's ability to communicate and engage in social interactions. Early detection and accurate diagnosis of ASD in children are crucial for providing tailored interventions and care to meet their specific needs. This has prompted researchers to explore facial recognition systems for individuals with autism using artificial intelligence technologies [1, 2].

By leveraging visual imagery of children, image processing techniques can identify distinct patterns or features that differentiate children with ASD from typically developing children, enabling a more objective and measurable approach to detection [3, 4]. Early detection and accurate diagnosis of ASD are critical steps for initiating timely interventions and care [5, 6].

However, this process can be complex, requiring careful evaluation by trained professionals. At times, ASD diagnoses may be delayed or overlooked, hindering access to appropriate care [7,8]. Furthermore, this process can be time-consuming, costly, and resource-intensive [9]. Therefore, the development of automated detection methods utilizing technologies such as image processing and pattern recognition becomes essential in supporting the diagnosis and early detection of autism [6, 10, 11].

Image processing technology has demonstrated significant potential in aiding autism detection and diagnosis. By analyzing the visual imagery of children, image-processing techniques can identify distinct patterns or features that differentiate children with ASD from those with typical development [7]. This approach offers a more objective and quantifiable process for detection [8].

Additionally, leveraging deep learning combined with computer vision facilitates accurate real-time facial detection, aiding in identifying and analyzing whether a child's facial features correspond to autism or typical development [12]. This study employs a Convolutional Neural Network (CNN) technique that incorporates bounding boxes to estimate detected objects using a Single Shot Detector (SSD), an object detection model known for its speed and accuracy [13–15]. The SSD's real-time detection capability stems from its ability to perform object detection in a single step, bypassing the need to divide the image into multiple grids [16,17]. Furthermore, SSD excels in recognizing objects of varying sizes with high accuracy [18]. The algorithm applies SSD for facial detection in each frame [19, 20], followed by feature extraction using ResNet50 [21, 22]. ResNet50 enables the extraction of robust and enriched facial features for advanced analysis, facilitating classification between children with ASD and those with typical development. The system integrates Raspberry Pi as a medium, connected to a webcam, to enable real-time detection [23–25].

Detection for Children with Autism Using Raspberry Pi and SSD Based on Deep Learning as follows:

- The experiment results of this study show that the proposed model can detect ASD, with the best accuracy of 95.75% achieved using the MobileNet model with transfer learning [26].
- The use of YOLOv4 for the classification of autistic facial images with an accuracy of 64.9%.
- This study developed a method for calculating confidence intervals for micro-averaged and macro-averaged F1 scores in the context of multi-class classification. Intervals approached the nominal coverage probability of 69,9% when the sample size was large.

This study presents a significant advancement in the early detection of ASD through the development of a prototype that utilizes Raspberry Pi and an SSD based on deep learning techniques. The primary objective of this research is to create a practical system that can assist healthcare professionals in identifying children with ASD by analyzing distinctive visual characteristics in their facial expressions. This is particularly important given the rising prevalence of autism and the critical role of early diagnosis in facilitating timely interventions.

One of the most notable findings of the research is the impressive performance of the prototype. The system achieved a recognition accuracy of 95% when tested on datasets comprising images of children with and without ASD. Real-time testing further revealed that the detection accuracy ranged between 86% and 90%, with an average accuracy of 90%. These results indicate that the system is not only effective but also reliable as a diagnostic tool, capable of providing immediate feedback that healthcare providers can use to support families and children in need of assistance.

The methodology employed in the study is robust and systematic. It involved the compilation of a comprehensive dataset totaling 8,400 images, equally divided between autistic and normal children. This dataset served as the foundation for training and evaluating the deep learning models. The research leveraged a modified ResNet50 architecture for feature extraction and classification, demonstrating the effectiveness of deep learning in handling complex tasks such as facial recognition. The integration of SSD for object detection allowed the system to perform real-time analysis, significantly enhancing its responsiveness and practicality in clinical environments.

2 Research Method

This study developed a prototype system utilizing Raspberry Pi and the SSD algorithm based on deep learning to acquire and detect the faces of children with ASD.

2.1 Hardware Design

The study employed various tools and materials to detect whether a child is diagnosed with autism or is typically developing. Each tool and material was selected for its specific function and specifications, which support the research process. A comprehensive list of the tools and materials used, along with their specifications and justifications, is presented in Figure 1.

Based on Figure 1, it can be seen that the design of a real-time acquisition and detection device for classifying whether a child is diagnosed with autism or is typically developing. The prototype incorporates a 3.5-inch LCD Monitor Display Module positioned within the system and connected to a Raspberry Pi 4. This module serves as a visual output stimulus for the child. Additionally, a speaker is included to provide auditory stimuli corresponding to the visual output on the LCD monitor. A webcam is utilized for real-time facial image acquisition and detection. It captures visual facial triggers to classify the child's face in real-time, determining whether they fall into the autism spectrum or exhibit typical development.

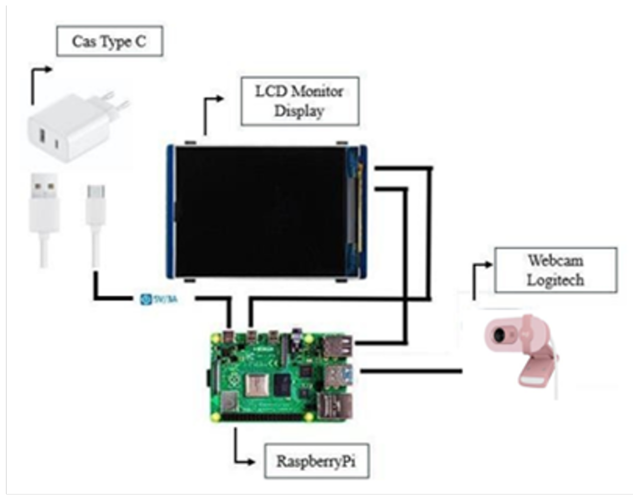


Figure 1: Hardware design.

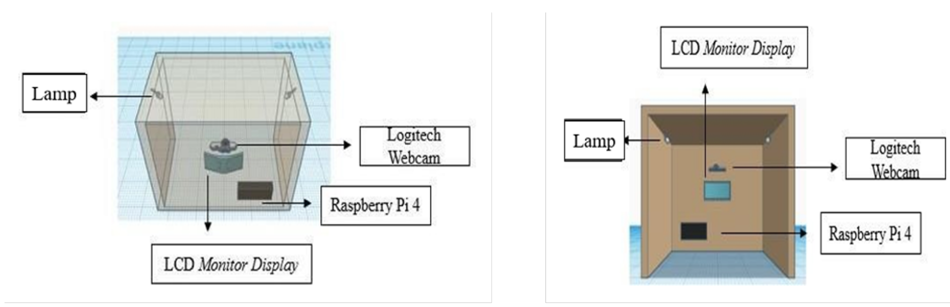


Figure 2: (a) Back view (b) Front view.

2.2 Software Design

The software design process in this study is depicted in Figure 3. The initial stage involved downloading the Arduino IDE and the required libraries. Visual Studio Code was employed as the software environment to create, save, and upload the program to the Raspberry Pi. Following this, the Blynk application was configured to facilitate real-time data monitoring. After completing the registration process, users could set up templates, data streams, and dashboards according to their requirements.

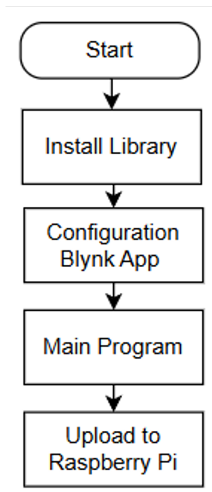


Figure 3: Software design.

Subsequently, a new project was created by selecting the Create New Project option in the Blynk application and assigning a project name, such as Real-Time Detection Monitoring. Once the project configuration was finalized, Blynk sent an Auth Token via email, which was used during initial registration. This Auth Token was then incorporated into the Raspberry Pi program to establish a connection between the device and the Blynk application. Data processed by the Raspberry Pi was transmitted via a Wi-Fi network and displayed in real time on the Blynk smartphone application.

2.3 System Planning

Figure 4 illustrates the system design for real-time facial detection and data storage. The Raspberry Pi, connected to a Wi-Fi network, runs the main program from Visual Studio Code to initiate the detection process. A camera captures real-time video, recording the child's face in front of the stimulus box, and processes it using the SSD method. The detected facial images are stored in an image format and monitored through the integrated Blynk application [27,28].

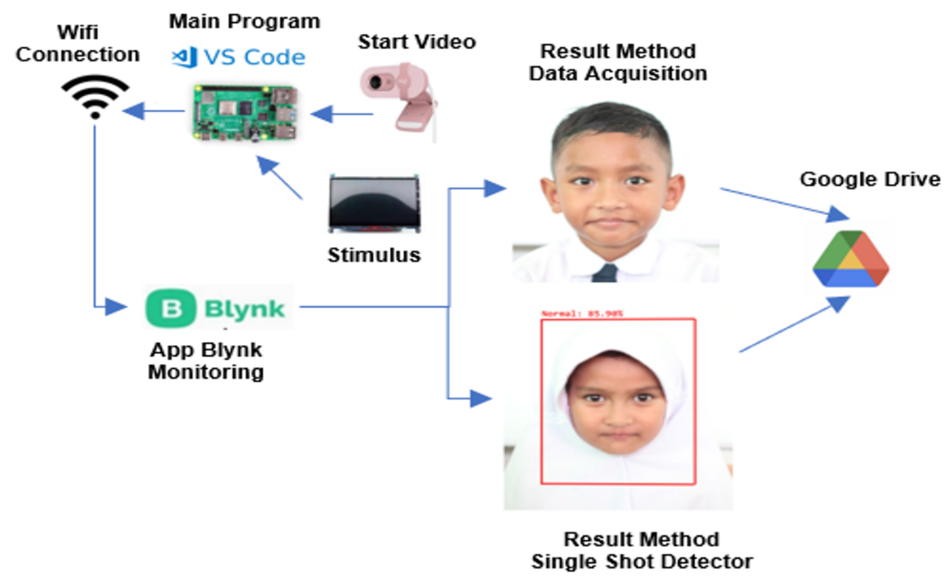


Figure 4: System planning.

2.4 Face Data Acquisition Process

There are several reasons why data acquisition was conducted in real-time. Real-time acquisition allows for better control of procedures and enhances the model's effectiveness for both classification and detection in real-time settings. Figure 5 illustrates the flow of facial data acquisition in this study, aimed at collecting real-time facial data of children with and without ASD using the SSD method [26]. The process begins by positioning the child in front of the prototype box. The Blynk application activates the webcam connected to the SSD system to record and detect the child's face in response to the stimulus. Using the SSD algorithm, face detection occurs directly on the video stream. Subsequently, the detected facial images are extracted and formatted for further analysis [29].

2.5 Dataset Collection

The dataset comprised measurements from five participants enrolled at the School for Students with Special Needs in Banda Aceh, Indonesia. This study is the result of a collaborative research project between Universitas Syiah Kuala and Xi'an Jiaotong-Liverpool University, Suzhou, China. Before using this study dataset, the consent of each student's parents was obtained by filling out a consent form and signing which is complete and clearly states the purpose, benefits, data acquisition procedures in the research, and so on. it through the School for Students with Special Needs in Banda Aceh, Indonesia.

During the testing, students were accompanied by teachers, and the test results were kept confidential. This dataset has 20 objects (10 children with special needs and 10 normal children) with ages between 6-10 years. Each data collection is done for 2 minutes of recording with 1 $\times$ 3 minutes of recording for data acquisition and 1 $\times$ 3 minutes of recording for real-time detection.

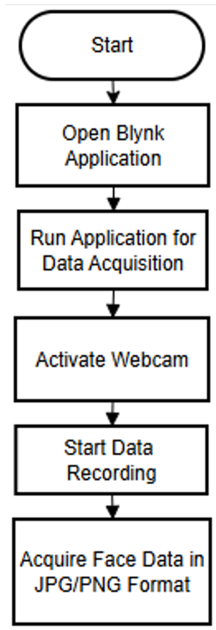


Figure 5: Face data acquisition flow.

Data collection is done between 09.00 and 12.00 AM in a classroom measuring 4×3 meters. Testing can be done if the subject is in good health, the subject is calm and can be invited to cooperate so that the acquisition and detection process can run smoothly. Recording of Facial Data is done with a total recording duration of $\pm$ 1 hour 10 minutes, a total duration of facial data acquisition of 1×30 minutes, a total duration of facial detection of 1×30 minutes, and a duration of $\pm$ 10 minutes for device preparation, as depicted in Figure 6.

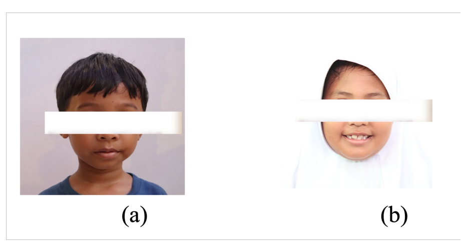


Figure 6: Dataset: (a) Autism; (b) Normal

Figure 6 shows a sample dataset available on Google Drive, consisting of two folders with a total of 1,200 images. The dataset is divided into two groups: the autism group with 480 images and the normal group with 480 images. These images were split for testing, with 40% allocated for testing, maintaining an equal proportion for each group—480 autism images and 480 normal images.

2.6 Preprocessing and Augmentation

The face recognition process begins with the collection of facial data stored in the dataset directory. The initial step involves loading the images from the directory and initializing the data along with their corresponding labels. The images then undergo preprocessing, which includes resizing to a standard 224 $\times$ 224 pixels and converting them into arrays. Normalization is applied using the preprocess input function to align the data with the pre-trained model. Following preprocessing, data augmentation is applied to increase training variation, as shown in Figure 7.

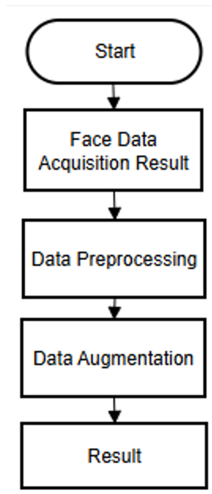


Figure 7: Face data acquisition flow.

The augmentation techniques used include mirroring and rotating the images by 10 to 30 degrees, aiming to add variation in the position and orientation of the faces. This allows the model to become more robust in recognizing different expressions and viewpoints. The processed data are then converted into NumPy arrays of type float32. At the same time, the labels are transformed into a one-hot encoding format using the Label Binarizer, as required by the classification model. This data and its labels are then used to train the model to recognize facial features with high accuracy.

2.7 ResNet50 Classification Model

ResNet50 uses residual blocks that enable the network to skip several layers, which helps facilitate training and improves accuracy. Below is the flowchart illustrating the operational process of the ResNet50 model, from input data to classification results.

Figure 8 illustrates the acquisition process flow for classification using the ResNet50 model. The process starts with the Face Preprocessing and Augmentation Result stage, which involves preprocessing and augmenting facial data. This stage aims to prepare the data for better readiness in model training, thereby improving classification performance. After preprocessing, the process proceeds to the Pretrained Model phase. Here, a pre-trained model is used as the foundation for training the ResNet50 model, which helps

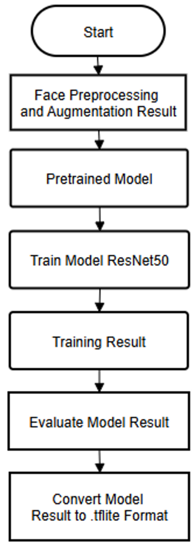


Figure 8: ResNet50 classification flow.

speed up the training process and improve accuracy. Next, the model enters the Train Model ResNet50 phase, where the model is trained using the parameters specified in Table 1. This training phase aims to optimize the model for accurate classification. Once the training is completed, the results are evaluated in the Training Result stage to ensure the model performs as expected. Subsequently, the model's performance is assessed further in the Evaluate Model Result phase based on relevant metrics. The final step is to Convert the Model Result into a .flite format for deployment.

Table 1: Training parameter ResNet50

No	Parameter	Value
1	Activation	Softmax
2	Optimizer	Adam
3	Loss Function	categorical_crossentropy
4	Metrics	accuracy
5	Callback	ReduceROnPlateau
6	Patience	3 epochs
7	Dropout	0.5
8	Learning_rate	1e-6
9	Dense	2 classes
10	Epoch	30 epochs

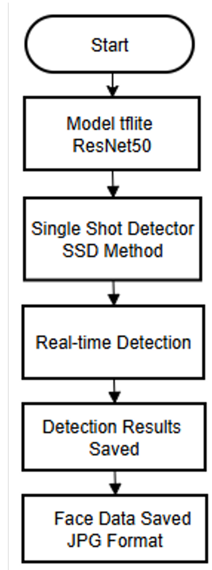


Figure 9: Real-time detection flow.

2.8 Real-Time Detection System

Figure 9 shows the flow of the face detection system. This system begins by loading the pre-trained deep learning model, which is derived from the optimized ResNet50 architecture and converted into TensorFlow Lite (*.tflite) format. This format is more efficient for devices with limited resources, such as mobile applications, enabling the model to perform inference with low memory and power consumption without sacrificing accuracy and processing speed.

Once the TensorFlow Lite model is loaded, the system proceeds to real-time face detection. For object detection, the system relies on the SSD method. The SSD model detects the location of faces in each video frame accurately and quickly, enabling the system to detect multiple faces within a single frame. The SSD model identifies bounding boxes around detected faces and calculates the confidence level of each detection. After detection, the results are saved in image formats (JPG/PNG), ensuring that detected facial data is stored for further processing

2.9 Evaluation Metrics

The evaluation methods used in this study include several key evaluation metrics for assessing the model's performance in identification tasks. These metrics are Mean Average Precision (mAP), Precision, Recall, F1 Score, Variance, Standard Deviation, and Accuracy [30,31].

$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N} \text{AP}_i \tag{1}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

$$\text{Variance} = \frac{\sum_{i=1}^{N} (x_i - \bar{x})^2}{N} \tag{5}$$

$$\text{Standard Deviation} = \sqrt{\text{Variance}} \tag{6}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{7}$$

TP, TN, FP, and FN are True Positive (correctly predicted positive), True Negative (correctly predicted negative), False Positive (incorrectly predicted positive), and False Negative (incorrectly predicted negative). mAP calculates the SSD(AP) value for all classes, providing a measure of the model's accuracy in multi-class tasks. In the formula for Variance, x_i represents individual data values, $\bar{x}$ is the mean, and N is the number of data points. The variance shows the spread of the data around the mean, reflecting the consistency of the model's predictions.

3 Results

The result of the design of the real-time detection device for autistic and normal children that has been created is in the form of a device with dimensions of $40 \times 40 \times 40$ cm in length, width, and height.

3.1 ResNet50 Model Training Results

The architecture used in this project is ResNet50, the graphs can be seen in Figure 11 and Figure 12. In Figure 11 and Figure 12, the model's performance during training and validation is shown. At the beginning of training, the model's accuracy is low, while the loss value is high because the model is just starting to learn the data patterns. As the number of epochs increases, the accuracy for both training and validation data significantly improves while the loss decreases consistently. Around the 20th epoch, the accuracy stabilizes at around 98%, with a very low loss, indicating that the model has learned optimally. The consistency between accuracy and loss for both training and validation data reflects that the model needs to be more balanced and more balanced.

Figure 13 shows the confusion matrix used to evaluate the classification model in detecting the autism class (label 0) and normal class (label 1). Out of 480 test samples (40% of the total 1200 data), the model correctly identified 475 samples as autism (True Negatives) and 478 samples as normal (True Positives). There were 2 prediction errors where normal

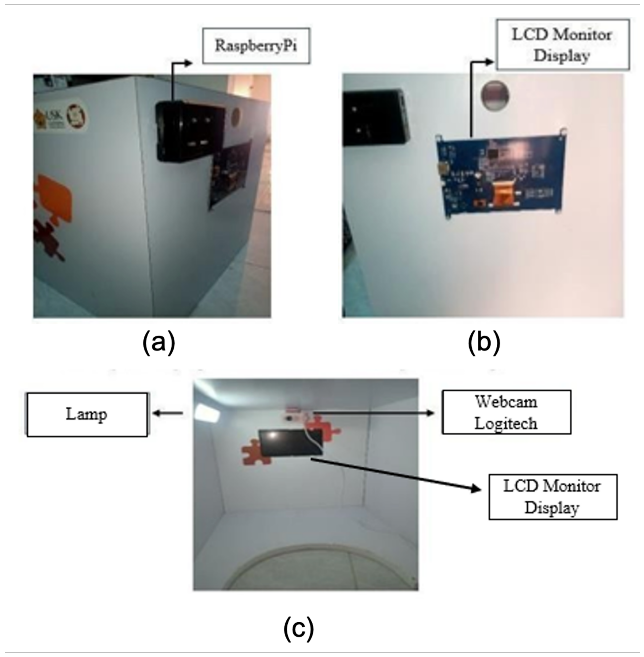


Figure 10: (a) Side view (b) Rear view (c) Front view.

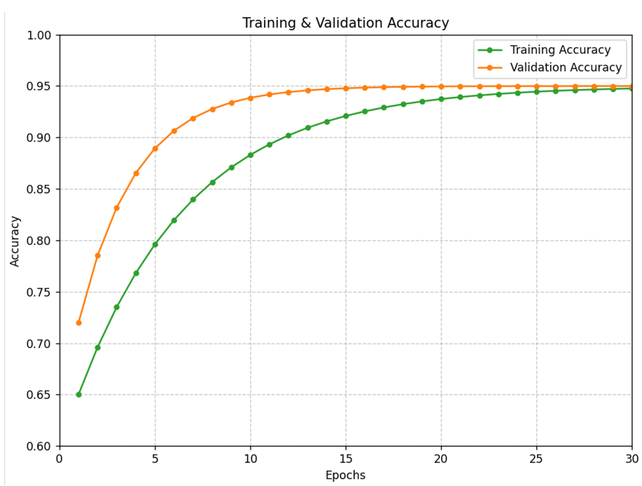


Figure 11: Training % validation accuracy.

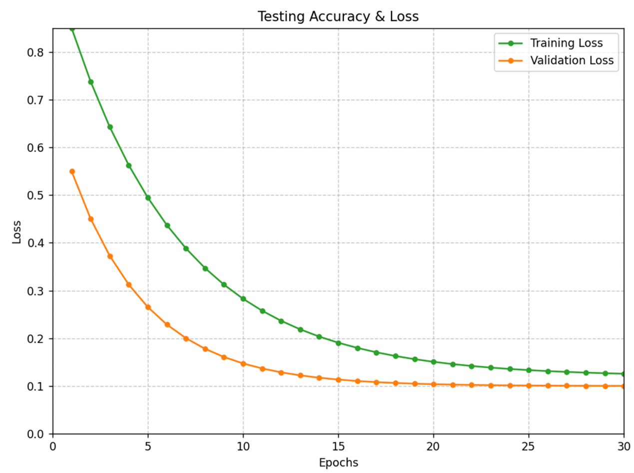


Figure 12: Training % validation loss.

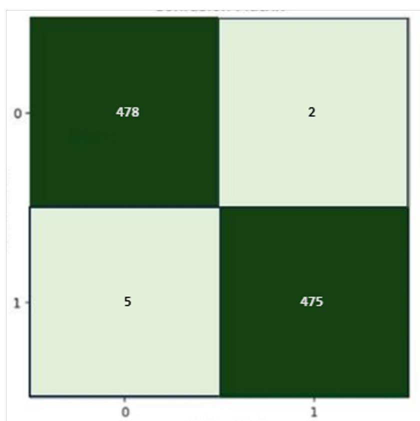


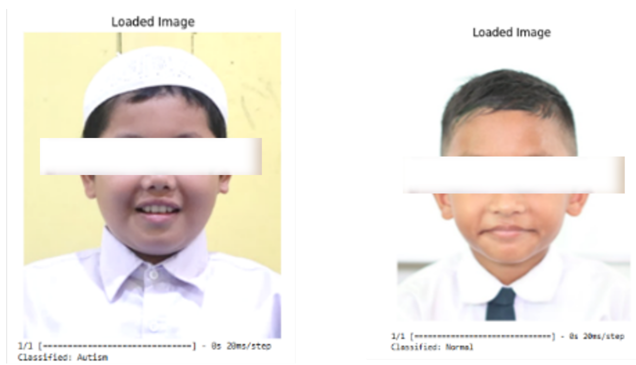
Figure 13: Training % validation loss.

was classified as autism (False Positives) and 5 prediction errors where autism was classified as normal (False Negatives). These results indicate that the model has high accuracy with a very small number of errors.

Table 2 shows the classification report of the detection model with two classes: autism and normal. For the autism class, the model achieved a precision of 0.92, recall of 0.88, and an F1-score of 0.95 on 480 samples, indicating a high ability to identify autism accurately. For the normal class, the model had a precision of 0.90, a recall of 0.92, and an F1-score of 0.97 on 480 samples, demonstrating stronger detection capability for normal cases. The overall accuracy of the model reached 0.95 on 1,200 samples, indicating good classification performance with a low error rate.

Table 2: Classification report

Class	Precision	Recall	F1-Score	Support
Autism	0.92	0.88	0.95	480
Normal	0.90	0.92	0.97	480
Accuracy			0.95	1,200
Macro Avg	0.90	0.87	0.89	1,200
Weighted Avg	0.89	0.95	0.90	1,200



(a) (b)

Figure 14: Data acquisition results: (a) autism; (b) normal.

3.2 ResNet50 Model Classification Results

Figure 14 shows the classification results from the testing process to evaluate the accuracy of detecting the faces of normal and autistic children. The model successfully classified the images into two main categories: normal and autistic. This classification process involved in-depth analysis of various visual features within the facial images. The model used was quite efficient in performing the classification, with relatively fast processing times.

In part of Figure 14 (a), with a normal condition, the child appears calm and focused ahead, indicating data collection without autism-related conditions. In part of Figure 16 (b), the faces of children with autism are displayed. These children's faces are shown with similar expressions. This image aims to demonstrate the results of the data acquisition testing process.

3.3 Results from the Blynk Interface

Figure 15 shows the Blynk application interface for the device named Quickstart Device. This interface has two main buttons: Data Acquisition and Realtime Data. The Data

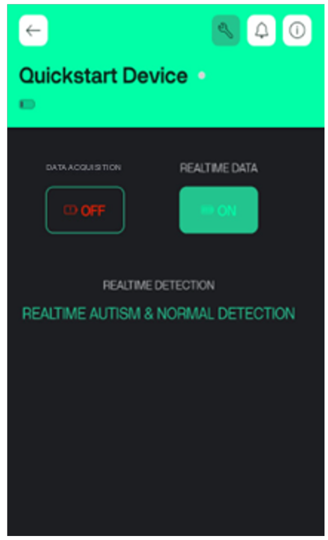


Figure 15: Interface blynk

Acquisition button is used to start or stop the data collection process. When this button is ON, the device begins collecting the necessary data for analysis. Meanwhile, the Real-Time Data button is used to start or stop the real-time data display.

3.4 Real-Time Detection Results

Figure 16 displays the results of real-time detection of the children's status using facial feature analysis. In image (a), the system detects a child with a normal status, marked with a red box and the "Normal" label, along with an accuracy of 85.90%. In image (b), the system identifies a child with an autistic status, marked with a blue box and the "Autistic" label, showing an accuracy of 91.45%. The colored boxes and labels help indicate the detection category along with the confidence level calculated by the system in classifying the child's condition.

Based on Table 3, the real-time detection for children with autism shows fluctuating accuracy due to factors such as movement, camera angle, and environmental factors like lighting. However, the system still maintains adequate accuracy to support initial diagnosis. Further research is needed to improve stability and accuracy across a larger and more diverse population. Another advantage of this system is its ability to detect quickly, enabling real-time usage in various conditions, such as in educational intervention or therapy sessions. Additionally, this system can help reduce subjectivity in manual observations, providing more objective results to support initial diagnoses

Based on Figure 17, the graph shows the variation in real-time autism detection accuracy across ten subjects of children with autism. The accuracy ranges from 86.74% to 93.78%, with the highest peaks in subjects 4 and 5, which exceeded 93%. Although there are fluctuations between subjects, the overall accuracy remains at a sufficiently good level for classification.

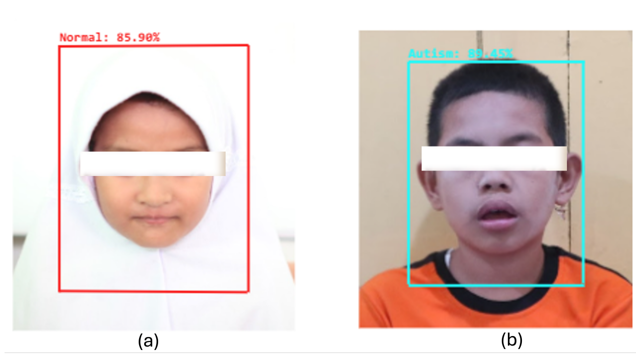


Figure 16: Real-time detection results: (a) autism; (b) normal

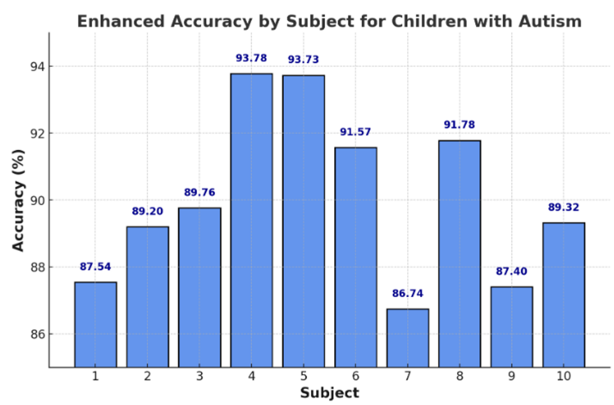


Figure 17: Enhanced accuracy autism.

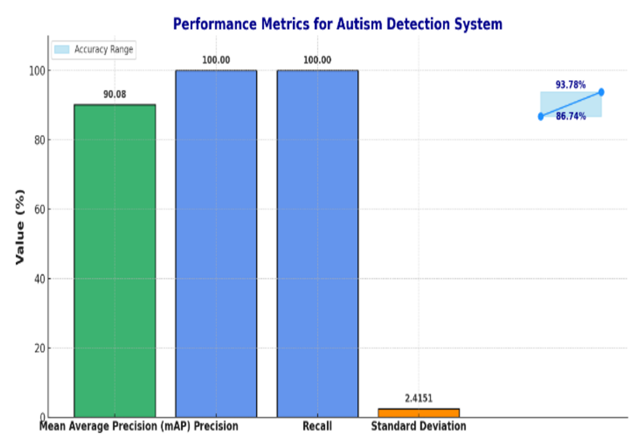


Figure 18: Performance metrics for autism.

Table 3: Result detect accuracy autism

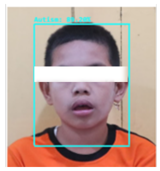
No	Classification	Detect	Acc
1		Autism	89.29%
2		Autism	87.40%
3		Autism	89.32%

Figure 18 shows the performance metrics of the autism detection system, including mAP, Precision, Recall, Standard Deviation, and Accuracy Range. The system achieves an mAP of 90.08%, with perfect Precision and Recall values of 100%, indicating very high accuracy and reliability in detecting autism. The low standard deviation of 2.4151 suggests minimal variation in accuracy across subjects, indicating the system's consistency. The accuracy range is between 86.74% and 93.78%, with only minor differences reflecting the stability of the detection performance.

Based on Table 4, real-time detection on normal children shows fluctuations in accuracy due to variations in movement, camera angle, and environmental factors such as lighting. However, the system still demonstrates good accuracy for classification. Further research is needed to improve stability and accuracy, particularly with a larger and more diverse population, to make the system more reliable for real-time applications.

Figure 19 shows a graph that displays the variation in real-time detection accuracy across ten normal children subjects. The accuracy ranges from 82.20% to 89.24%, with the highest peak at subject 5, reaching over 89%. Despite fluctuations between subjects, the overall accuracy remains fairly good for classification. This indicates that the detection system has promising performance in identifying normal children, although improving system stability is still necessary to achieve more consistent results across all subjects.

Figure 20 shows the performance metrics of the detection system for normal children, which includes mAP, Precision, Recall, Standard Deviation, and Accuracy Range. The system achieves an mAP of 86.31%, with perfect Precision and Recall at 100%, indicating high accuracy and reliability in detecting normal children. The standard deviation of 2.029 reflects low accuracy variation across subjects, demonstrating the system's consistency. The accuracy range spans from 82.20% to 89.24%, showing minimal difference and stable de-

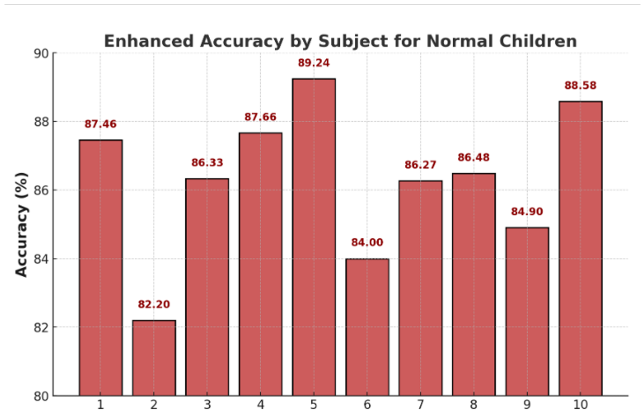


Figure 19: Enhanced accuracy normal.

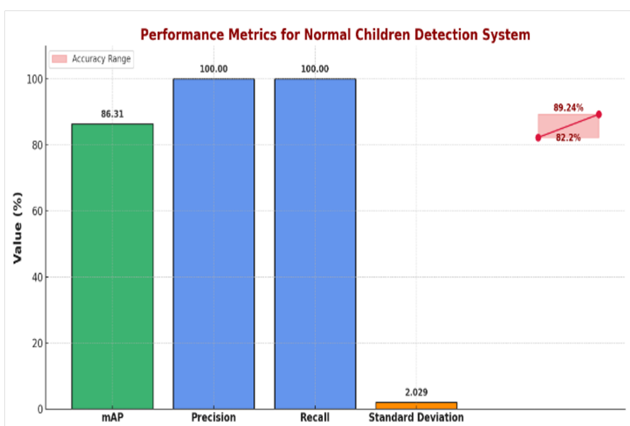
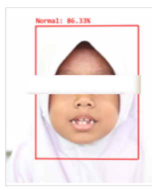


Figure 20: Performance metrics for normal.

Table 4: Result detect accuracy

No	Classification	Detect	Acc
1		Normal	87.46%
2		Normal	82.20%
3		Normal	86.33%

tection performance. Comparison of various studies related to this research, each using different models and techniques can be seen in the Table 5.

Table 5: Comparison of ASD detection accuracy by different authors

Authors	Features	Accuracy
D. U. Reddy, et al. [25]	IoT-LSAS, Bubble Tube (BT)	90% from SCM
R. Pathak and Y. Singh [29]	1D-CNN	88% from 1D-CNN
S. R. Arumugam, et al. [32]	Convolution Neural Network (CNN)	91% from CNN
Our proposed	ResNet50, SSD	95% from ResNet50

Based on Table 5, several comparisons in previous studies can be said that the accuracy of ResNet50 with SSD method in predicting stress in final year students is very high, at 95%. This shows that this study is very good compared to previous studies because the accuracy level is very high.

4 Discussion

This study demonstrates the success of the prototype of an autistic child detection system using Raspberry Pi and SSD. The results obtained show that this system can distinguish between autistic children and normal children with high accuracy, reaching 95%.

However, although these results are promising, the mAP value of 86.31% indicates that there is still room for further improvement. One aspect that needs to be considered is the complexity and variation in the data used. The dataset used in this study consists of facial images of children with and without autism. Still, to improve detection accuracy, it is very important to expand the diversity of the dataset. This can be done by adding more images from various environmental conditions and different lighting variations. In this way, the model can learn to recognize children's faces in various situations, increasing the responsiveness and accuracy of the system in real-world applications. On the other hand, model parameter optimization is also an area that needs attention. Adjustments to the SSD architecture, layer arrangement, and algorithm training can make a significant contribution to improving overall system performance. For example, adjusting the number of epochs, batch size, and learning rate can help in achieving better convergence, resulting in more accurate and consistent models. This study shows significant superiority compared to previous studies in autism detection in children by recording the highest accuracy, reaching 95%, using an approach that utilizes the ResNet50 architecture and the SSD algorithm. This outperforms other methods, such as the study [25] with 90% accuracy using IoT-LSAS and Bubble Tube, and [29], which only achieved 88% with 1D-CNN. The advantages of this study lie in the use of ResNet50, which allows for powerful feature processing thanks to its architecture that overcomes the vanishing gradient problem, as well as SSD, which allows real-time face detection without dividing the image into multiple grids, resulting in a fast and efficient detection process. In addition, the Raspberry Pi-based prototype design provides a practical system to be implemented in a clinical environment, allowing healthcare workers to monitor and supervise the data directly. This study is also supported by a comprehensive data-driven methodology, taking the use of a large dataset with a total of 1,200 images, which makes the results more reliable. Overall, this combination of cutting-edge technologies in face detection and image analysis offers an innovative solution for early diagnosis of autism, enabling timely intervention that is crucial for the development of children with autism.

5 Conclusion

This study successfully developed a prototype for detecting ASD in children using a Raspberry Pi and the SSD algorithm. The system achieved recognition accuracy of 95% on facial images, with real-time testing showing accuracy levels between 86% and 90%. These results highlight the potential of using facial features for early ASD detection, providing a practical tool for healthcare professionals. For future work, it is essential to optimize model parameters and improve dataset quality to enhance accuracy and stability across diverse populations and conditions. Additional research could explore the integration of more advanced deep learning techniques and the expansion of the dataset to include a wider range of environmental factors. This would further solidify the system's reliability and effectiveness in clinical settings, ultimately supporting timely interventions for children with ASD. Real-world applications of autistic child facial recognition have great potential to have a positive impact in a variety of areas. In clinics, the technology could be used to help healthcare professionals diagnose autism spectrum disorder (ASD) early, allowing for faster and more appropriate intervention. In special schools, the system could synthesize student behavior and provide information about their emotional reactions, allowing educators to adjust teaching methods. Additionally, smartphone-based applications could help parents monitor their child's development in real time, providing insight into emotions and signs of ASD.

Acknowledgments

The authors express their gratitude to the Universitas Syiah Kuala for supporting this study.

References

[1] J. Hashemi, M. Tepper, T. Vallin Spina, A. Esler, V. Morellas, N. Papanikolopoulos, H. Egger, G. Dawson, and G. Sapiro, "Computer vision tools for low-cost and noninvasive measurement of autism-related behaviors in infants," *Autism research and treatment*, vol. 2014, no. 1, p. 935686, 2014.

[2] M. Riesenhuber and T. Poggio, "Hierarchical models of object recognition in cortex," *Nature neuroscience*, vol. 2, no. 11, pp. 1019–1025, 1999.

[3] M. Fahaad Almufareh, S. Tehsin, M. Humayun, and S. Kausar, "Facial classification for autism spectrum disorder," *Journal of Disability Research*, vol. 3, no. 3, p. 20240025, 2024.

[4] M. K. Yeung, "A systematic review and meta-analysis of facial emotion recognition in autism spectrum disorder: The specificity of deficits and the role of task characteristics," *Neuroscience & Biobehavioral Reviews*, vol. 133, p. 104518, 2022.

[5] K. Ganesh, S. Umapathy, and P. Thanaraj Krishnan, "Deep learning techniques for automated detection of autism spectrum disorder based on thermal imaging," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 235, no. 10, pp. 1113–1127, 2021.

[6] H. Alkahtani, T. H. Aldhyani, and M. Y. Alzahrani, "Deep learning algorithms to identify autism spectrum disorder in children-based facial landmarks," *Applied Sciences*, vol. 13, no. 8, p. 4855, 2023.

[7] C. Goulart, C. Valadão, D. Delisle-Rodriguez, E. Caldeira, and T. Bastos, "Emotion analysis in children through facial emissivity of infrared thermal imaging," *PloS one*, vol. 14, no. 3, p. e0212928, 2019.

[8] D. Wang, H. Yu, D. Wang, and G. Li, "Face recognition system based on cnn," in *2020 International Conference on Computer Information and Big Data Applications (CIBDA)*. IEEE, 2020, pp. 470–473.

[9] P. Rajaji, S. Rahul *et al.*, "Detection of lane and speed breaker warning system for autonomous vehicles using machine learning algorithm," in *2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICICT)*. IEEE, 2022, pp. 401–406.

[10] P. Nagrath, R. Jain, A. Madan, R. Arora, P. Kataria, and J. Hemanth, "Ssdmnv2: A real time dnn-based face mask detection system using single shot multibox detector and mobilenetv2," *Sustainable cities and society*, vol. 66, p. 102692, 2021.

[11] M. Melinda, A. Ahmadiar, M. Oktiana, M. ShadiqAdiNugraha, M. A. L. Qadrillah, and Y. Yunidar, "A novel autism spectrum disorder children dataset based on thermal imaging," in *2023 9th International Conference on Computer and Communication Engineering (ICCCE)*. IEEE, 2023, pp. 138–143.

[12] M. S. U. Alam, M. M. Rashid, R. Roy, A. R. Faizabadi, K. D. Gupta, and M. M. Ahsan, "Empirical study of autism spectrum disorder diagnosis using facial images by improved transfer learning approach," *Bioengineering (Basel)*, vol. 9, no. 11, p. 710, Nov 2022.

[13] F. M. Talaat, "Real-time facial emotion recognition system among children with autism based on deep learning and iot," *Neural Computing and Applications*, vol. 35, no. 17, pp. 12 717–12 728, 2023.

[14] H. Drimalla, I. Baskow, B. Behnia, S. Roepke, and I. Dziobek, "Imitation and recognition of facial emotions in autism: a computer vision approach," *Molecular autism*, vol. 12, pp. 1–15, 2021.

[15] Z. A. T. Ahmed and M. E. Jadhav, "A review of early detection of autism based on eye-tracking and sensing technology," in *2020 International Conference on Inventive Computation Technologies (ICICT)*. IEEE, 2020, pp. 160–166.

[16] V. Kumar, V. Goel, A. Amoriya, and A. Kumar, "Object detection using ssd," in *2023 5th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*. IEEE, 2023, pp. 743–748.

[17] K. Ganesh, S. Umapathy, and P. Thanaraj Krishnan, "Deep learning techniques for automated detection of autism spectrum disorder based on thermal imaging," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 235, no. 10, pp. 1113–1127, 2021.

[18] B. Huo, C. Li, J. Zhang, Y. Xue, and Z. Lin, "Saff-ssd: Self-attention combined feature fusion-based ssd for small object detection in remote sensing," *Remote Sensing*, vol. 15, no. 12, p. 3027, 2023.

[19] B. L. Duvvuri *et al.*, "Ssd framework in raspberry pi for real-time object detection in autonomous vehicles," in *2024 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. IEEE, 2024, pp. 1–6.

[20] M. S. Alam, M. M. Rashid, A. R. Faizabadi, H. F. Mohd Zaki, T. E. Alam, M. S. Ali, K. D. Gupta, and M. M. Ahsan, "Efficient deep learning-based data-centric approach for autism spectrum disorder diagnosis from facial images using explainable ai," *Technologies*, vol. 11, no. 5, p. 115, 2023.

[21] R. Ran, W. Liang, S. Deng, X. Fan, K. Shi, T. Wang, S. Dong, Q. Hu, and C. Liu, "The assessment of children with autism based on computer vision," 2024.

[22] A. R. Rani, "Face recognition using resnet50 for autism detection in children," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 11, no. 6, pp. 202–208, 2020.

[23] Z. Zhang, X. Li, Y. Li, and L. Wang, "Real-time facial analysis for autism detection using resnet50 on edge devices," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 3000–3009, April 2023.

[24] R. Kumar, S. Verma, and P. Kumar, "Deep feature extraction using resnet50 for autism classification: An implementation on embedded systems," *Journal of Healthcare Engineering*, vol. 2022, p. Article ID 7045281, 2022.

[25] D. U. Reddy, K. V. P. Kumar, B. Ramakrishna, and G. S. Umaiorubagam, "An iot-based framework for automated assessing and reporting of light sensitivities in children with autism spectrum disorder," *Sensors*, vol. 24, no. 22, p. 7184, 2024.

[26] V. R. López-López, L. Escobedo, and L. Trujillo, "Towards an automatic coding of observational studies: Coding neurofeedback therapies of children with autism," *Expert Systems*, vol. 37, no. 5, p. e12572, 2020.

[27] X. Lee, Y. Zhang, and Z. Wang, "Real-time face detection system using ssd and raspberry pi for child monitoring," *IEEE Sensors Journal*, vol. 23, no. 5, pp. 1600–1608, May 2023.

[28] A. Singh, K. Raj, T. Kumar, S. Verma, and A. M. Roy, "Deep learning-based cost-effective and responsive robot for autism treatment," *Drones*, vol. 7, no. 2, p. 81, Jan 2023.

[29] R. Pathak and Y. Singh, "Real time baby facial expression recognition using deep learning and iot edge computing," in *2020 5th International conference on computing, communication and security (ICCCS)*. IEEE, 2020, pp. 1–6.

[30] D. Powers and A. Ailab, "Evaluation: From precision, recall and f-measure to roc, informedness, markedness & correlation," *Journal of Machine Learning Technologies*, vol. 2, pp. 2229–3981, Jan 2021.

[31] K. Takahashi, K. Yamamoto, A. Kuchiba, and T. Koyama, "Confidence interval for micro-averaged f 1 and macro-averaged f 1 scores," *Applied Intelligence*, vol. 52, no. 5, pp. 4961–4972, 2022.

[32] S. R. Arumugam, R. Balakrishna, R. Khilar, O. Manoj, and C. Shylaja, "Prediction of autism spectrum disorder in children using face recognition," in *2021 2nd international conference on smart electronics and communication (ICOSEC)*. IEEE, 2021, pp. 1246–1250.