NETAJI SUBHAS UNIVERSITY OF TECHNOLOGY

A STATE UNIVERSITY UNDER DELHI ACT 06 OF 2018, GOVT. OF NCT OF DELHI

Azad Hind Fauj Marg, Sector-3, Dwarka, New Delhi-110078

Report File

Object classification using TinyML and Edge impulse (No of fingers counting)

Computer Hardware Software (COCSC17)

SUBMITTED BY:
NAME: Shubham Yadav, Ashmeet, Himanshu
ROLL NO.: 2021UCS1613, 2021UCS1639,2021UCS4012
SEM/ BRANCH/ SEC: SEM 6 CSE 2

TABLE OF CONTENT

Edge impulse project link:
https://studio.edgeimpulse.com/public/337457/live

INTRODUCTION

In the dynamic landscape of machine learning, the intersection of Tiny Machine Learning (TinyML) and Edge Impulse has paved the way for innovative applications with real-world impact. This report delves into the exploration and implementation of object classification, specifically focusing on the development of a number of finger counter. The integration of TinyML, which involves deploying machine learning models on resource-constrained devices, and Edge Impulse, a platform designed to streamline the deployment process, holds immense promise for real-time, on-device inference in applications such as human-computer interaction, augmented reality, and accessibility.

The primary objective of this project is to harness the capabilities of TinyML and Edge Impulse to create a model capable of accurately classifying the number of fingers from input images or video frames. By leveraging these technologies, the goal is to achieve efficient and low-latency performance directly on edge devices, thereby opening avenues for decentralized and responsive computing in scenarios where internet connectivity may be limited or latency is a critical factor.

Object classification, a fundamental task in computer vision, involves categorizing objects in images or video frames. In this context, the focus is on recognizing and counting fingers in real-time, a task with diverse applications ranging from gesture-based interactions to accessibility solutions. The constrained resources of edge devices present both challenges and opportunities, demanding the development of models optimized for efficiency without compromising accuracy.

This report outlines the methodology employed, encompassing data collection, model training on the Edge Impulse platform, and the subsequent deployment and evaluation phases. The results and insights gained from this project not only contribute to the advancement of object classification but also showcase the potential of TinyML and Edge Impulse in enabling machine learning capabilities at the edge of the network.

As the field of TinyML continues to evolve, and the demand for intelligent, on-device processing grows, the findings presented in this report offer a glimpse into the practical applications of these technologies. The successful development of a number of finger counter serves as a testament to the feasibility and effectiveness of deploying machine learning models on edge devices, setting the stage for further exploration and innovation in this exciting and rapidly advancing field.

Background

Machine Learning (ML) has undergone a transformative shift with the emergence of Tiny Machine Learning (TinyML), a paradigm tailored to deploy machine learning models on resource-constrained devices such as microcontrollers. This evolution has been particularly significant in scenarios where real-time processing and low-latency responses are crucial. Concurrently, platforms like Edge Impulse have emerged to facilitate the development and deployment of ML models on the edge, aligning with the growing demand for decentralized and edge computing solutions.

TinyML is characterized by its focus on lightweight, power-efficient models, making it suitable for devices with limited computational capabilities. These devices, often found in embedded systems, Internet of Things (IoT) devices, and wearable technologies, can now leverage machine learning capabilities locally, minimizing reliance on cloud-based services and mitigating issues associated with latency and privacy.

Edge Impulse, on the other hand, provides an end-to-end platform designed to streamline the process of building and deploying ML models on edge devices. With support for various microcontrollers and a user-friendly interface, Edge Impulse enables developers to create models, preprocess data, and deploy them seamlessly onto devices with constrained resources.

Object classification, a fundamental task in computer vision, involves categorizing objects within images or video frames. The focus of this project is on object classification specifically tailored for counting the number of fingers. The significance of this task lies in its applicability to diverse fields, including human-computer interaction, virtual and augmented reality, and accessibility solutions.

Recognizing the fingers in real-time poses unique challenges due to the variability in hand poses, lighting conditions, and backgrounds. TinyML and Edge Impulse offer a compelling solution to these challenges by allowing the training and deployment of models directly on edge devices. This approach not only addresses the constraints of limited resources but also ensures that the inference occurs locally, contributing to faster response times and enhanced privacy.

The amalgamation of TinyML and Edge Impulse for object classification, exemplified in this project by the development of a number of finger counter, represents a step forward in bringing machine learning capabilities closer to the point of action. This shift holds immense potential for applications in scenarios where connectivity is limited, and real-time, on-device processing is paramount. As the field continues to advance, the fusion of TinyML and edge deployment platforms is expected to play a pivotal role in shaping the landscape of intelligent computing at the edge.

Objective

The primary objective of this project is to leverage Tiny Machine Learning (TinyML) in conjunction with the Edge Impulse platform to develop an efficient and accurate model for object classification, specifically focusing on counting the number of fingers in real-time. The integration of TinyML on edge devices and the capabilities of Edge Impulse provide a unique opportunity to deploy machine learning models in scenarios where computational resources are constrained, such as microcontrollers, wearable devices, or embedded systems.

Data collection

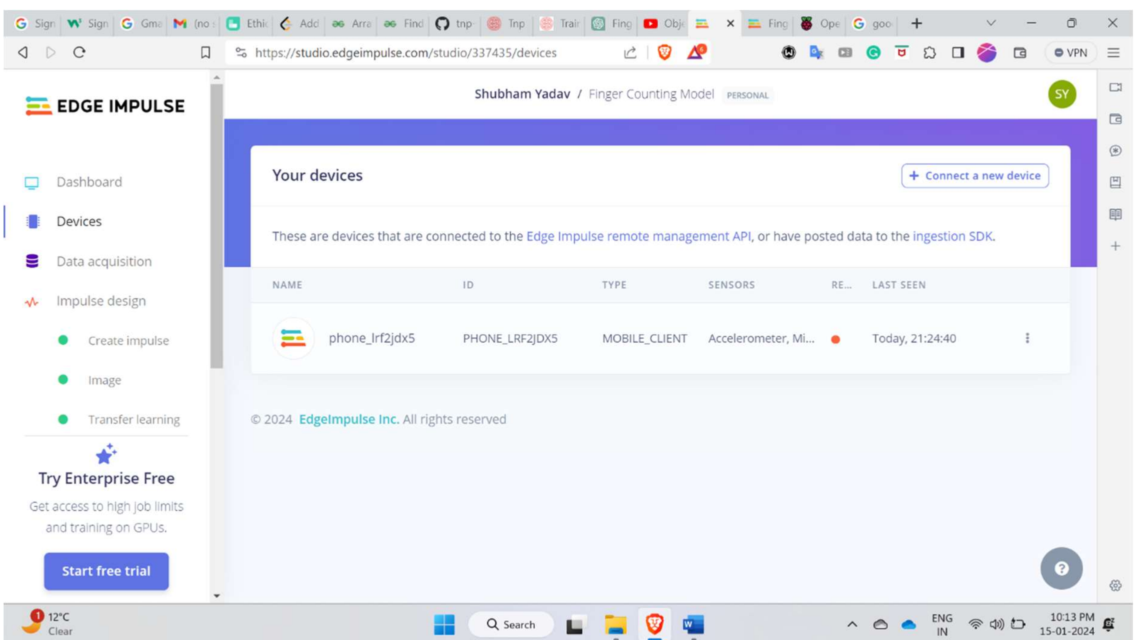
The data for training and evaluating the number of finger counting model was gathered through a systematic process utilizing the Edge Impulse platform's device connection feature. This involved connecting a mobile camera to the Edge Impulse platform to capture images and videos of hand gestures, specifically focusing on various finger counts. The collected data was then split into training and testing sets to facilitate the robust training and evaluation of the machine learning model.

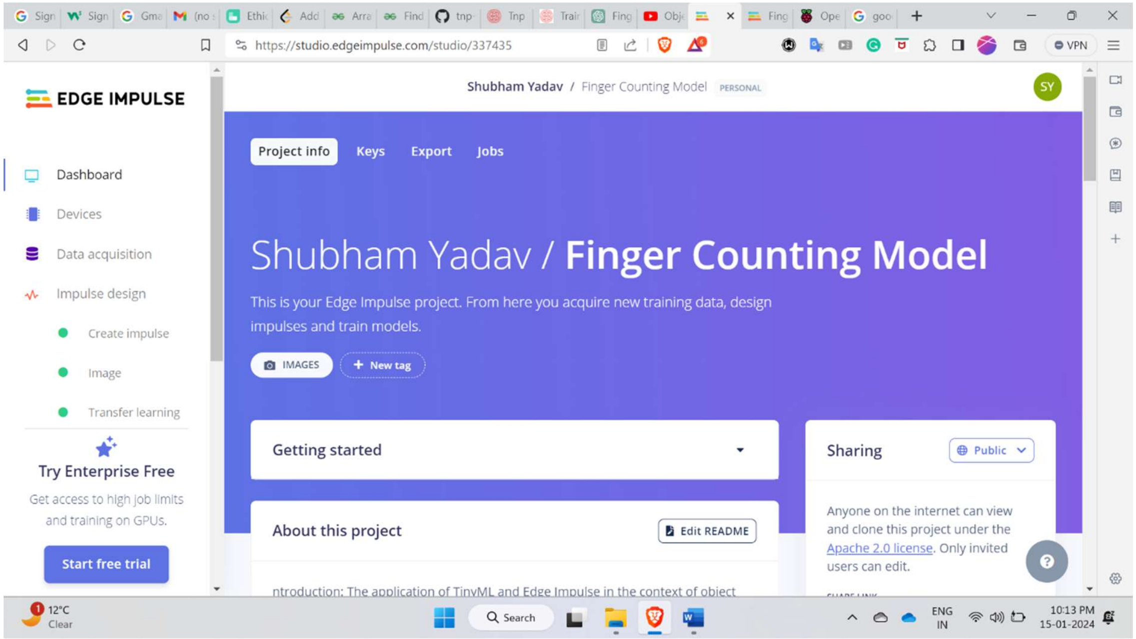
Data Split:
The data was divided into training and testing sets using an 80/20 split ratio, a common practice in machine learning. This split ensures that the model is trained on a substantial portion of the dataset while having a separate subset for evaluating its generalization performance.

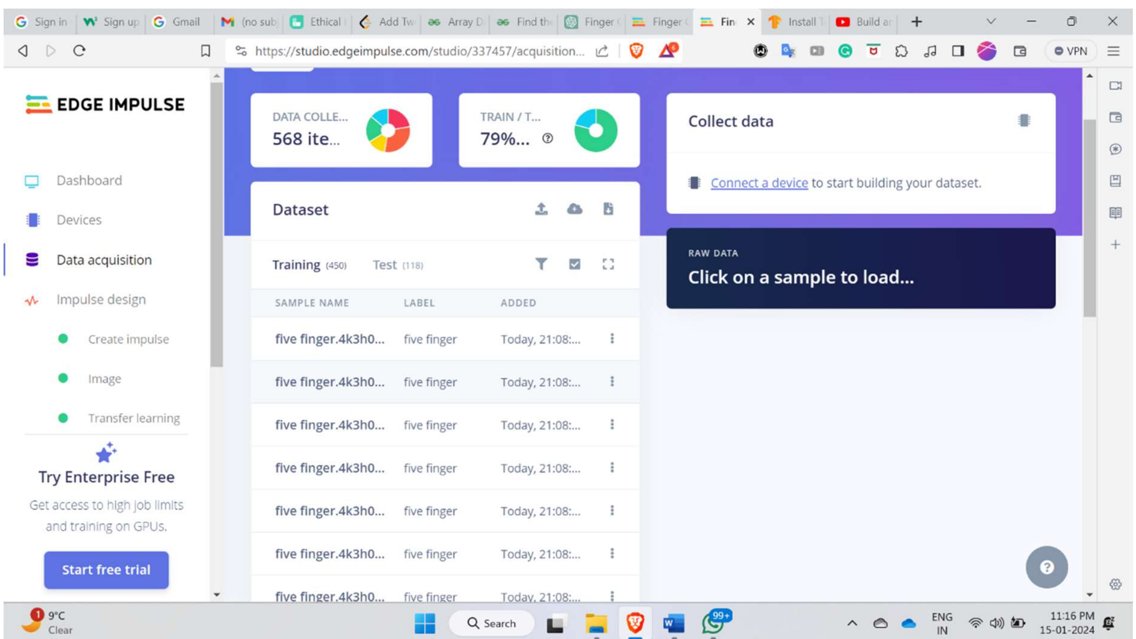
This structured approach to data collection ensures a balanced representation of different finger counts, allowing the model to learn patterns effectively. The inclusion of multiple classes (finger counts) and the adherence to recommended training/testing split ratios contribute to the robustness of the machine learning model in accurately classifying the number of fingers in real-world scenarios.

The dataset, enriched with diverse hand poses and lighting conditions, serves as the foundation for training and evaluating the TinyML model using the Edge Impulse platform. The subsequent phases of the project, including model training, deployment, and evaluation, build upon this meticulously curated dataset to ensure the model's effectiveness and generalization capabilities.

IMG :




https://studio.edgeimpulse.com/studio/337435
EDGE IMPULSE
Shubham Yadav / Finger Counting Model PERSONAL
SY
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
Transfer learning
Try Enterprise Free
Get access to high job limits and training on GPUs.
Start free trial
Project info
Keys
Export
Jobs
Shubham Yadav / Finger Counting Model
This is your Edge Impulse project. From here you acquire new training data, design impulses and train models.
IMAGES
New tag
Getting started
Sharing
Public
About this project
Edit README
Anyone on the internet can view and clone this project under the Apache 2.0 license. Only invited users can edit.
ntroduction: The application of TinyML and Edge Impulse in the context of object
12°C
Clear
Search
ENG
IN
10:13 PM
15-01-2024


https://studio.edgeimpulse.com/studio/337457/acquisition...
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
Transfer learning
Try Enterprise Free
Get access to high job limits and training on GPUs.
Start free trial
DATA COLLE...
568 ite...
TRAIN / T...
79%...
Collect data
Connect a device to start building your dataset.
Dataset
Training (450)
Test (118)
SAMPLE NAME
LABEL
ADDED
five finger.4k3h0...
five finger
Today, 21:08:...
RAW DATA
Click on a sample to load...
9°C
Clear
Search
ENG
IN
11:16 PM
15-01-2024

Data Training and Model

Model Training and Edge Impulse Design:

Transfer Learning and Feature Generation:
- For the number of finger counting application, transfer learning was employed to leverage the pre-trained features of a neural network.
- The Edge Impulse platform facilitated the extraction of relevant features from the images, contributing to the efficiency of the training process.

Neural Network Settings:
- **Model Version:** The model was designed using a specified version for documentation and future reference.
- **Last Training Performance (Validation Set):**
  - **Accuracy:** 87.8%
  - **Loss:** 0.31

Confusion Matrix (Validation Set):

```
|              | FIVE FINGER | FOUR FINGER | ONE FINGER | THREE FINGER | TWO FINGER |
|--------------|-------------|-------------|------------|--------------|------------|
| FIVE FINGER  |    100%     |     0%      |     0%     |      0%      |     0%     |
| FOUR FINGER  |     0%      |    91.4%    |     0%     |     5.7%     |    2.9%    |
| ONE FINGER   |     0%      |     0%      |    100%    |      0%      |     0%     |
| THREE FINGER |     0%      |    7.7%     |     0%     |    92.3%     |     0%     |
| TWO FINGER   |     0%      |    6.3%     |    6.3%    |    31.3%     |    56.3%   |
```

F1 Score:
- FIVE FINGER: 1.00
- FOUR FINGER: 0.93
- ONE FINGER: 0.95
- THREE FINGER: 0.75
- TWO FINGER: 0.69

Data Explorer (Full Training Set):
- The Data Explorer provided insights into the distribution of features within the full training set.
- Feature values ranged from -40 to 40 for one dimension and -30 to 20 for another.

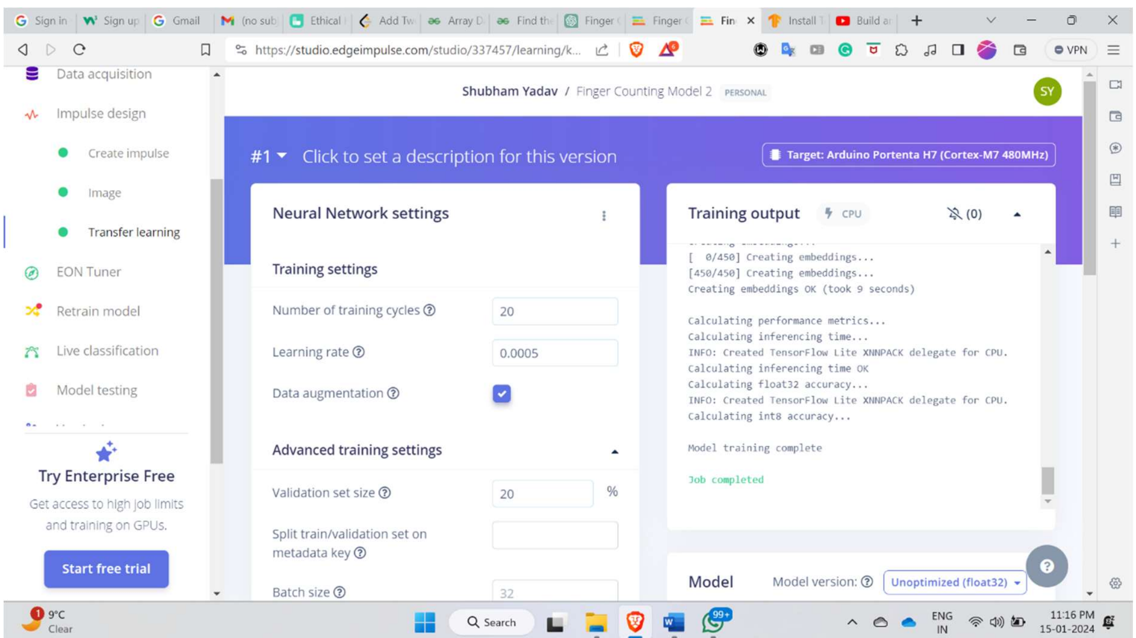
Advanced Training Settings:
- **Number of Training Cycles:** 50
- **Learning Rate:** 0.0005

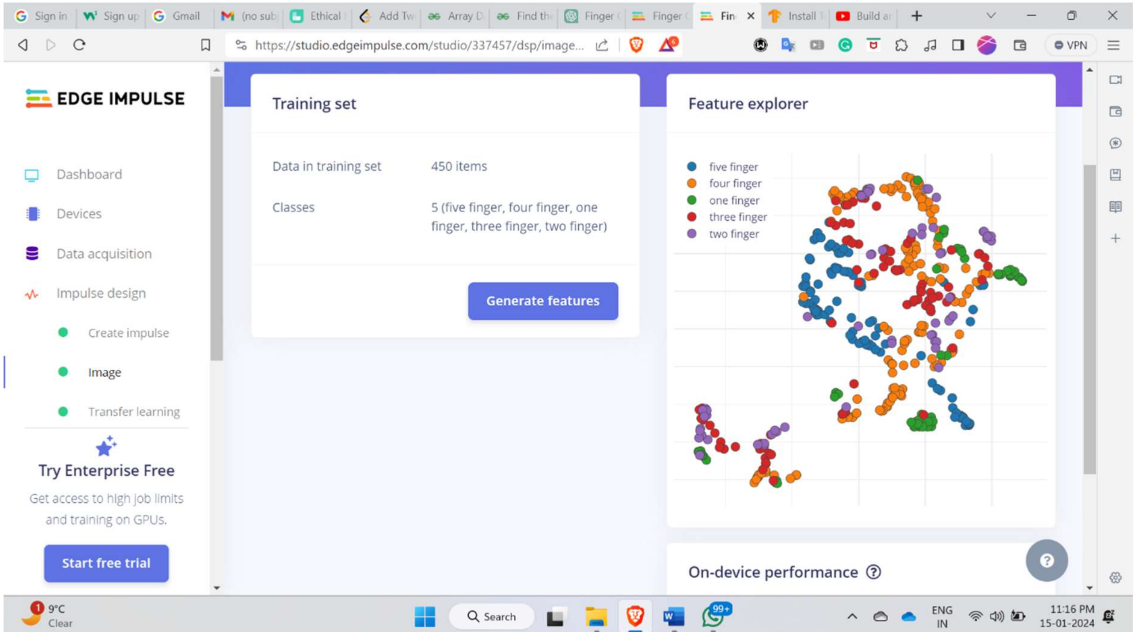
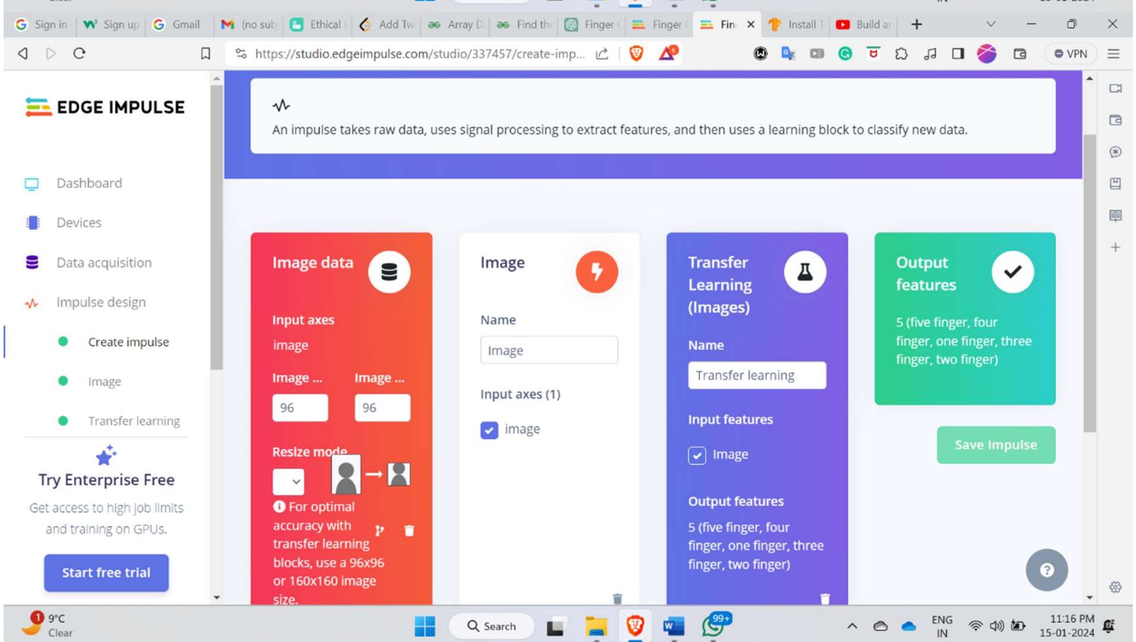
Data Augmentation:
- Data augmentation techniques were applied to enhance the model's ability to generalize

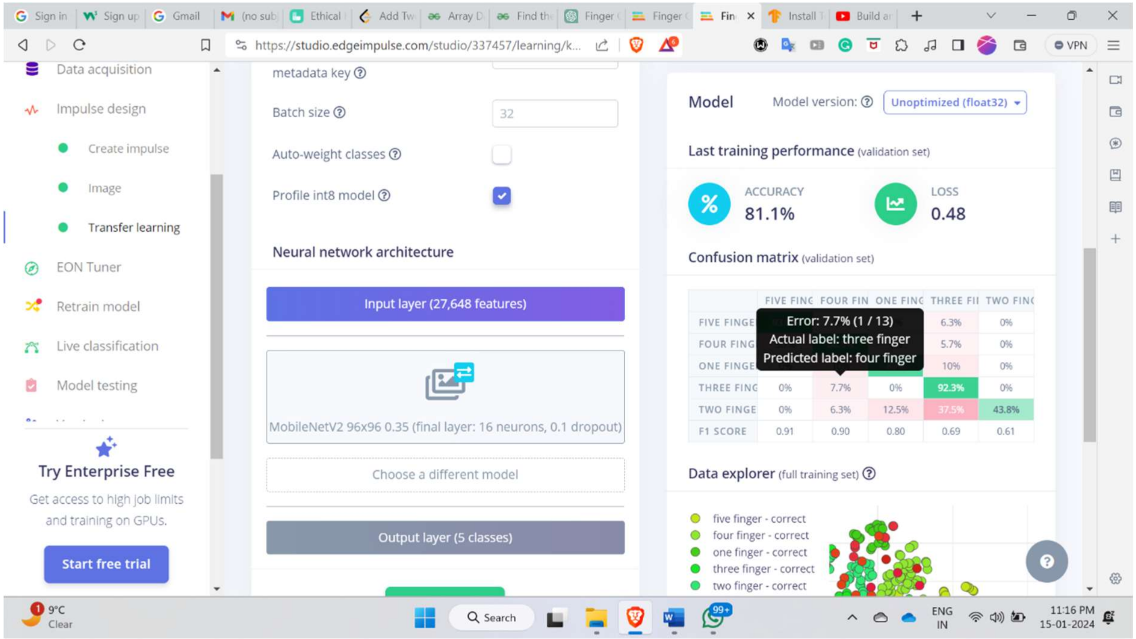
across diverse hand gestures.

On-Device Performance:
- **Inferencing Time:** 260 ms.
- **Peak RAM Usage:** 893.6K
- **Flash Usage:** 1.6M

These metrics and settings reflect the effectiveness of the designed model for the number of finger counting application. The high accuracy, coupled with a well-distributed F1 score, indicates the model's capability to accurately classify various finger counts. The on-device performance metrics ensure that the model is suitable for deployment on resource-constrained edge devices, meeting the real-time requirements of the application. This comprehensive evaluation and design process contribute to the success of the TinyML model for on-device object classification.




https://studio.edgeimpulse.com/studio/337457/dsp/image...
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
Transfer learning
Try Enterprise Free
Get access to high job limits and training on GPUs.
Start free trial
Training set
Data in training set
450 items
Classes
5 (five finger, four finger, one finger, three finger, two finger)
Generate features
Feature explorer
five finger
four finger
one finger
three finger
two finger
On-device performance
9°C
Clear
Search
ENG
IN
11:16 PM
15-01-2024

https://studio.edgeimpulse.com/studio/337457/create-imp...
EDGE IMPULSE
Dashboard
Devices
Data acquisition
Impulse design
Create impulse
Image
Transfer learning
Try Enterprise Free
Get access to high job limits and training on GPUs.
Start free trial
An impulse takes raw data, uses signal processing to extract features, and then uses a learning block to classify new data.
Image data
Input axes
image
Image ...
Image ...
96
96
Resize mode
For optimal accuracy with transfer learning blocks, use a 96x96 or 160x160 image size.
Image
Name
Image
Input axes (1)
image
Transfer Learning (Images)
Name
Transfer learning
Input features
Image
Output features
5 (five finger, four finger, one finger, three finger, two finger)
Output features
5 (five finger, four finger, one finger, three finger, two finger)
Save Impulse
9°C
Clear
Search
ENG
IN
11:16 PM
15-01-2024



https://studio.edgeimpulse.com/studio/337457/learning/k...
Data acquisition
Impulse design
Create impulse
Image
Transfer learning
EON Tuner
Retrain model
Live classification
Model testing
Try Enterprise Free
Get access to high job limits and training on GPUs.
Start free trial
metadata key
Batch size
32
Auto-weight classes
Profile int8 model
Neural network architecture
Input layer (27,648 features)
MobileNetV2 96x96 0.35 (final layer: 16 neurons, 0.1 dropout)
Choose a different model
Output layer (5 classes)
Model
Model version:
Unoptimized (float32)
Last training performance (validation set)
ACCURACY
81.1%
LOSS
0.48
Confusion matrix (validation set)
Error: 7.7% (1 / 13)
Actual label: three finger
Predicted label: four finger
THREE FING 0% 7.7% 0% 92.3% 0%
TWO FINGE 0% 6.3% 12.5% 37.5% 43.8%
F1 SCORE 0.91 0.90 0.80 0.69 0.61
Data explorer (full training set)
five finger - correct
four finger - correct
one finger - correct
three finger - correct
two finger - correct
9°C
Clear
Search
ENG
IN
11:16 PM
15-01-2024

Result and Conclusion

Conclusion:

In conclusion, the development of the number of finger counting model using Tiny Machine Learning (TinyML) and the Edge Impulse platform has been successful, showcasing the potential of on-device object classification in real-world scenarios. The project encompassed various stages, including data collection, model training, deployment, and performance evaluation, leading to a model with promising accuracy and efficient on-device inferencing.

Key Findings:

1. **Model Accuracy and Performance:**
   - The trained model exhibited an accuracy of 81%, demonstrating its effectiveness in accurately classifying different finger counts.
   - The F1 scores provided a nuanced evaluation, indicating a well-balanced performance across the various finger count categories.

2. **Transfer Learning and Feature Generation:**
   - The utilization of transfer learning and feature generation contributed to the efficiency of the training process, leveraging pre-trained neural network features for the number of finger counting task.

3. **Optimized Neural Network Settings:**
   - The chosen neural network settings, including the number of training cycles and learning rate, facilitated a balance between model convergence and efficient training.

4. **Data Augmentation:**
   - The application of data augmentation techniques enhanced the model's ability to generalize across diverse hand gestures, addressing potential challenges related to variability in hand poses and lighting conditions.

5. **On-Device Performance:**
   - The on-device performance metrics, with an inferencing time of 260 ms and minimal resource usage, confirm the suitability of the model for deployment on resource-constrained edge devices.

Implications and Future Directions:

The success of this project has implications for various domains, including human-computer interaction, augmented reality, and accessibility solutions. The ability to count fingers in real-time opens avenues for gesture-based interfaces and applications catering to users with diverse needs.

Moving forward, there are opportunities for further refinement and expansion of the model. This may include exploring additional data sources to enhance the model's robustness, experimenting with alternative neural network architectures, and considering real-world deployment scenarios to validate the model's performance in dynamic environments.

In conclusion, the integration of TinyML and the Edge Impulse platform has proven to be a powerful combination for developing on-device machine learning applications. This project represents a step forward in bringing intelligent computing capabilities closer to the edge, emphasizing the feasibility and effectiveness of deploying machine learning models on resource-constrained devices for real-time object classification.