virtual notepad

Pushpa kumar
23b0983
MENTOR : shorya sethia

July 14, 2024

Contents

1 objective

This project's main goal is to develop a virtual notepad that uses hand gestures captured by a webcam to enable users to write, erase, and predict handwritten characters. In order to provide a seamless drawing and writing experience, this project makes use of computer vision and machine learning techniques to recognize and respond to hand gestures.

2 Learning

2.1 week1

- Learnt how to use jupyter lab and google colab
- Basics of python(loops,recursive functions,data structures,class and objects etc).
- learnt modules like numpy,matplot,pandas etc.
- Introduction to ML

Introduction to ML

What is Machine Learning?
Machine learning is a field of artificial intelligence that involves the development of algorithms that allow computers to learn and make predictions or decisions based on data.
Types of Machine Learning

- **Supervised Learning:** Algorithms learn from labeled training data, making predictions based on input-output pairs.
- **Unsupervised Learning:** Algorithms identify patterns and relationships in unlabeled data without predefined outputs.
- **Reinforcement Learning:** Algorithms learn by interacting with an environment, receiving rewards or penalties based on their actions.

conclusion
The first week laid a solid foundation, preparing for more advanced topics and applications in Python and ML.
Assignments

- some functions and graphs

2.2 week2

- Learnt about linear regression and logistic regression.
- learnt how to use tensorflow
- learnt how to overcome overfit(add batch normalization and dropouts)

linear regression:
The definition of linear regression is a supervised learning algorithm that fits a linear equation to observed data to model the relationship between a dependent variable and one or more independent variables. Continuous outcomes are predicted by it.
How it Works:

- $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n$
- y = dependent variable (output)
- β_0 = intercept
- $\beta_1, \beta_2, ..., \beta_n$ = coefficients
- $x_1, x_2, ..., x_n$ = independent variables (features)

Logistic Regression:
For binary classification issues, supervised learning algorithms like logistic regression are employed. By converting the linear combination of input features into a value between 0 and 1, which represents the probability of the target class, it models the probability of a binary outcome using a logistic function.By fitting a linear equation to the observed data, linear regression is a supervised learning algorithm that models the relationship between a dependent variable and one or more independent variables. It forecasts ongoing results.
How it Works:

- Formula: $P(y = 1) = \frac{1}{1+e^{-(\beta_0+\beta_1 x_1+\beta_2 x_2+...+\beta_n x_n)}}$
- $P(y = 1)$ = probability of the target class being 1
- β_0 = intercept
- $\beta_1, \beta_2, ..., \beta_n$ = coefficients
- $x_1, x_2, ..., x_n$ = independent variables (features)
- e = base of the natural logarithm (approximately 2.71828)
- These formulas illustrate how linear and logistic regression create models to predict outcomes based on input features.

neural networks
One class of machine learning algorithms that draws inspiration from the composition and operations of the human brain is called Neural Networks. They are made up of layers upon layers of networked nodes, or neurons, that process incoming data and develop prediction skills.

1. **Neurons:**Basic units of a neural network, similar to biological neurons. Each neuron receives input, applies a weight, adds a bias, and passes the result through an activation function.

2. **Layers:**

    - Input Layer: The first layer that receives the input features.
    - Hidden Layers: Intermediate layers where computation and feature extraction occur.
    - Output Layer: The final layer that produces the prediction or classification.

Convolutional Neural Networks (CNNs):

- Specialized for processing grid-like data such as images. They use convolutional layers to automatically and adaptively learn spatial hierarchies of features.

- Components: Convolutional layers, pooling layers, fully connected layers.

conclusion
learnt about ML and how to implement using tensorflow.
assignment
implementing back propagation of neural network without tensorflow.

2.3 week3

- mostly assignment

assignment

- Made cnn architecture for CIFAR100 from scratch.

- used pre-trained INCEPTIONV3 for CIFAR10

- Made Pneumonia or Normal classifier based on Chest X-Ray from kaggle x-ray dataset.

conclusion
Understood how to load data ,how to use pre-trained model and mostly how to implement CNN to predict.

2.4 week4

- Learnt about opencv.
- Learnt about mediapipe module

Made hand-detection module which detects 21 points on hand , so that it can be used in different code. **conclusion**
understood how to use mediapipe in opencv.

2.5 week5

- mostly assignment

assignment

- Made brightness control of the device from python programme using mediapipe and opencv.
- Made sound control of the device from python programme using mediapipe and opencv.
- Made a programme which pauses web cam if we place more than 3 fingers in front of webcam.
- Made a programme which acts like drag and drop images with fingures.

conclusion
Learnt how to use mediapipe in programming and had fun with assignments.

2.6 week6

Final project

3 Project

3.1 Modules used

1. **OpenCV:** For capturing video from the webcam and processing images.
2. **MediaPipe:** For hand detection and tracking.
3. **TensorFlow and Keras:** For loading pre-trained models to predict handwritten characters.
4. **Numpy:** For handling image and numerical data.
5. **Math:** For calculations related to drawing.
6. **Time:** For timestamping saved files.

3.2 Key Features:

1. Hand Gesture Detection:
    - Uses MediaPipe to detect and track hand positions.
    - Differentiates between index and middle finger positions to trigger various actions.
    - All selections are done if both middle and index finger is open.
    - After selecting draw or erase is done by closing middle finger and index finger open.
2. Drawing Functionality:
    - Users can draw on the canvas by selecting colors (red, green, white, blue).
    - A virtual paintbrush is controlled by the index finger.
3. Erasing Functionality:
    - Users can erase parts of the drawing by selecting the eraser icon.
    - Erasing is controlled by the index finger, which creates a "hole" effect on the canvas.
4. Character Prediction:
    - The model predicts handwritten characters drawn on the virtual canvas.
    - Six pre-trained models are used to improve the accuracy through a majority voting system.
5. User Interface:
    - An intuitive UI that allows users to select colors, eraser, and trigger prediction.
    - A dropdown menu for additional actions such as saving and exiting.
6. Saving and Clearing:
    - Users can save the current state of the canvas as an image file.
    - The canvas can be cleared to start a new drawing.

3.3 Implementation Details:

1. Drawing Lines:
    - The draw-line function draws lines on the canvas between two points.

- The distance between consecutive points is checked to ensure smooth drawing.

2. Erasing Areas:
   - The erase-area function creates a circular mask to erase parts of the drawing.
3. Preprocessing Images:
   - The preprocess-image function processes the drawn image to prepare it for model prediction.
   - Images are resized, converted to grayscale, and normalized.
4. Predicting Letters:
   - Six models predict the drawn character, and a majority vote determines the final prediction.
5. User Interaction:
   - Hand gestures are used to select drawing colors, erase, and predict.
   - A dropdown menu is activated using right-click and offers additional functionalities.

Results and Usage:
The virtual notepad successfully allows users to draw and erase using hand gestures detected by a webcam. The character prediction feature provides accurate results, making it a useful tool for digital note-taking and drawing. The user-friendly interface ensures easy interaction and a seamless experience.

4 Usage Instructions for Virtual Notepad with Hand Gesture Recognition

Prerequisites:

- Ensure you have the required libraries installed
- pip install opencv-python mediapipe tensorflow numpy

Running the Application:

1. Start the Application:
   - Run the script in your Python environment.

- The application will access your default webcam and display the virtual notepad interface.

User Interface Overview:

- The interface consists of a black bar at the top with color selection icons, an eraser icon, and a dropdown menu.
- The main drawing area is displayed below the black bar.
- Hand gestures are used to interact with the notepad.

Hand Gestures and Actions:

1. Drawing on the Canvas:
   - **Select a Color:** Use the index and middle fingers to point at the desired color icon on the top bar. Close the middle finger and use only the index finger to draw.
     - **Red Color:** Move the middle finger to the red icon area (rightmost color icon).
     - **Green Color:** Move the middle finger to the green icon area.
     - **White Color:** Move the middle finger to the white icon area.
     - **Blue Color:** Move the middle finger to the blue icon area (leftmost color icon).
   - **Draw:** After selecting a color, close the middle finger and use the index finger to draw on the canvas.
2. Erasing the Drawing:
   - **Select the Eraser:** Point at the eraser icon using the index and middle fingers. Close the middle finger and use only the index finger to erase.
   - **Erase:** Move the index finger over the area you want to erase.
3. Predicting Characters:
   - **Select Predict:** Point at the "PREDICT" text on the top bar using the index and middle fingers.
   - **Predict:** The application will process the drawing on the canvas and display the predicted character(s) at the bottom of the screen.
4. Clearing the Canvas:
   - **Select Clear:** Point at the "CLEAR" text on the top bar using the index and middle fingers.
   - The canvas will be cleared, and you can start drawing again.

5. Using the Dropdown Menu:
    - **Open the Menu:** Right-click on the "more" icon (three dots) at the top right corner of the screen.
    - Select an Option:
        - **Save:** Click on "save" to save the current canvas as an image file.
        - **Exit:** Click on "exit" to close the application.
        - Help: Click on "help" for instructions

Additional Notes:

- **Saving the Canvas:** When you save the canvas, an image file with a timestamp will be created in the current directory.
- Exiting the Application: You can also press the 'q' key to exit the application.