**Project Proposal: Building a CLIP-based Training Dataset for Badminton Stroke Classification** CLIP (Contrastive Language-Image Pretraining).

**approach:**

1.data collection

2.data preprocessing

3.data creating

4data training

5.data evaluation

Video Preprocessing

- Extract key frames representing different strokes (e.g., smash, shot, clear) using deep learning libraries like PyTorch or TensorFlow.

- Preprocess the extracted frames using OpenCV or ResNet libraries to enhance their quality and standardize dimensions.

import cv2

import torch

**def preprocess\_video(video\_path, sampling\_rate=5, target\_width=224, target\_height=224):**

**cap = cv2.VideoCapture(video\_path)**

**preprocessed\_frames = []**

**frame\_count = 0**

**while cap.isOpened():**

**ret, frame = cap.read()**

**if not ret:**

**break**

**if frame\_count % sampling\_rate == 0:**

**# Resize frame**

**frame = cv2.resize(frame, (target\_width, target\_height))**

**preprocessed\_frames.append(frame)**

**frame\_count += 1**

**cap.release()**

**return torch.tensor(preprocessed\_frames)**

**(after preprocessing we will convert features to tensor using pytorch)**

Dataset Creation:

- Organize the preprocessed frames into a structured dataset.

- Create a JSON dataset mapping each image to its corresponding label (stroke type).

- Store the dataset in MongoDB for scalability and ease of updating.

4. Model Training:

- Utilize machine learning models compatible with CLIP, such as PyTorch's CLIP model.

- Train the model on the constructed dataset, minimizing the loss function to optimize classification accuracy.

5. Evaluation:

- Assess the model's performance on a separate validation set to ensure generalization.

- Measure classification accuracy, precision, recall, and F1 score to gauge model effectiveness.