DAYANANDA SAGAR UNIVERSITY

SCHOOL OF ENGINEERING

COGNITIVE LOAD QUANTIFICATION FOR LEARNING ASSESSMENT

Project Implementation Details & Discussion

By

MEENAKSHI

[ENG24CSE0013]

M.TECH 3rd Semester,

Department of Computer Science & Engineering

DAYANANDA SAGAR UNIVERSITY

28-9-2025

CONTENTS

COGNITIVE LOAD QUANTIFICATION FOR LEARNING ASSESSMENT

Based On Computer Vision

1. Project Implementation Summary

This project establishes a novel, non-intrusive framework to quantify a student's **Cognitive Load (CL)** in real-time, serving as the basis for assessing learning and understanding capabilities defined by **Bloom's Taxonomy** levels.

I. Constructive Description

The foundational challenge is to achieve reliable CL quantification using a

non-intrusive Computer Vision (CV) methodology. The system's assessment of the learning state is dependent on the student's mood, cognitive load, and commitment. The core technical constraint is deploying a highly accurate model that is small (under 5MB) and capable of running **offline within a browser environment (JavaScript/FaceApi.js)**, eliminating dependency on constant streaming.

The methodology assumes a student's commitment based on time spent on the platform. The project then focuses on using CV as a proxy for physiological metrics (such as blood flow changes) by measuring key behavioral cues like head position, eye position, and posture, to

accurately quantify CL. The system will ultimately support on-demand question generation and answer evaluation using a deployed LLM.

2. Implementation

II. Implementation Phases

The entire project flow is driven by **evidence and validation**, aiming to determine whether CL or other factors (age, gender) are the primary differentiators in learning outcomes based on a structured **Training → Revision → MCQ** protocol.

Phase	Goal	Key Technical Focus	Justification
Phase 1: Advanced Feature Engineering	Replace raw pixel data with highly efficient, numerical geometric features for CL assessment.	**Feature Extraction (Python):** Use **MediaPipe** to detect facial and pose landmarks. Calculate Head Pose (Pitch, Yaw, Roll), Eye Aspect Ratio (EAR), and Body Angle to quantify distraction and engagement.	Creates the small, numerical input required for browser-based deployment and quantifies the physical state associated with CL .
Phase 2: Teacher Model Fusion & Distillation	Achieve high accuracy under the severe **<5MB size constraint** for browser execution.	**Knowledge Distillation:** Train a small, lightweight **Student Model** (e.g., a simple MLP) using the predictions of the large, high-accuracy **Teacher Model** (the previously trained ensemble/VGG16 model).	Solves the deployment challenge, allowing the model to run offline and quickly on mobile/laptops.
Phase 3: Deployment, Integration & Validation	Deploy the small CL model to the browser and build the final evaluation module.	**Browser Deployment:** Convert the Student Model to **TensorFlow.js** format. **Application Logic:** Integrate with **FaceApi.js** for live, client-side assessment and link to Bloom's Taxonomy levels.	Provides the final application and enables **wholly evidence-driven validation** of the CL metric against actual test performance.