Project 11: Computer Vision

Objective (Why?)

Build an image recognition system using OpenCV and PyTorch. This project teaches you to work with images and create AI that can "see" and classify objects. You will practice:

- Image Processing: Loading, resizing, and filtering images with OpenCV
- Simple Deep Learning: Using pre-trained models for image classification
- Web Interface: Creating an app where users can upload images
- Practical AI: Building something users can actually use

Core Requirements

Component	Requirement
Image Processing	OpenCV for loading, resizing, and basic image operations
Image Classification	Use pre-trained model (ResNet/EfficientNet) for object recognition
Web Interface	Streamlit app for image upload and predictions
Simple Training	Fine-tune model on small custom dataset (optional)

Development Approach: Milestone-Based Progression

Philosophy: Focus on deliverable quality and comprehensive review compliance rather than rigid timelines. Each milestone must pass all relevant review templates before proceeding.

Milestone 1: OpenCV Foundation & Image Processing

Deliverables:

- OpenCV installation and comprehensive image loading capabilities
- Image manipulation functions (loading, resizing, displaying)
- Filter applications (blur, sharpen, edge detection) with parameter tuning
- Multi-format image support with proper error handling
- Image processing pipeline with optimization and validation

Review Requirements (Must Pass to Proceed):

- Architecture Review: Clean image processing pipeline design
- Performance Review: Efficient image operations and memory management
- Security Review: Safe image file handling and validation

Milestone 2: Pre-trained Model Integration & Classification

Deliverables:

- PyTorch installation and pre-trained ResNet model loading
- Object classification system for common objects with confidence scoring
- Multiple image processing with batch operations and result aggregation
- Confidence score interpretation and prediction quality assessment
- Comprehensive error handling for various image types and edge cases

Review Requirements (Must Pass to Proceed):

- AI Integration Review: Effective model usage and classification accuracy
- Performance Review: Fast inference and efficient model operations
- Architecture Review: Clean model integration and result processing

Milestone 3: Web Interface & Production Features

Deliverables:

- Streamlit application with intuitive image upload interface
- Integrated image processing and classification pipeline
- Results display with confidence scores and visual feedback
- Multiple image support with batch processing capabilities
- User testing with non-technical users and usability improvements

Review Requirements:

- AI Integration Review: Complete computer vision application
- Architecture Review: Production-ready application architecture
- Code Quality Review: Clean code with comprehensive documentation
- Performance Review: Responsive user interface and processing speed

Milestone Progression Rules:

- Cannot advance to next milestone without passing all review requirements
- Flexible timing allows for learning at individual pace
- Quality gates ensure each milestone meets professional standards
- Mentor support available for concept clarification and review failures

Simplified Architecture

None

User uploads image → OpenCV preprocessing → PyTorch prediction → Web display

What Students Build

- Upload any image and get object classification
- See confidence scores for predictions
- View original vs processed images
- Simple, clean web interface anyone can use