Lab 03: 2025 Computer Vision with Classical Machine Learning

ITAI 1378 - Module 03 Lab Assignment

📋 Lab Overview

Welcome to your first hands-on computer vision project! In this lab, you will build face recognition models using classical machine learning techniques. More importantly, you'll learn why models with 99% training accuracy often fail in production and how to build models that actually work in the real world.

🎯 Learning Objectives

By completing this lab, you will:

1.Understand the complete classical ML pipeline for computer vision

2.Extract and compare different feature types (HOG, LBP, etc.)

3.Implement multiple ML algorithms and understand their strengths

4. 🚨 CRITICALLY IMPORTANT: Learn how training approaches affect model performance

5. 🚨 AVOID THE TRAP: Understand why 99% training accuracy often means model failure

6.Build models that work in the real world, not just in notebooks

📁 File Naming Convention

IMPORTANT: Your submission must follow this exact naming format:

Plain Text

L03_[YourLastName][YourFirstName]_ITAI_1378.ipynb

Examples:

•L03_SmithJohn_ITAI_1378.ipynb

•L03_GarciaMaría_ITAI_1378.ipynb

•L03_ChenWei_ITAI_1378.ipynb

⚠️ Submissions with incorrect naming will receive a 10-point deduction.

💻 Technical Requirements

Environment Options (Choose ONE):

1.Google Colab (Recommended for beginners)

•Free, cloud-based, no setup required

•Upload the notebook to Colab and run

2.Kaggle Notebooks

•Free, cloud-based alternative

3.Local Jupyter

•If you have Python and Jupyter installed locally

4.AWS SageMaker Studio Lab

•Free tier available for students

Required Libraries (Pre-installed in cloud environments):

•numpy, matplotlib, seaborn

•scikit-learn, scikit-image

•opencv-python-headless

📚 Lab Structure & Tasks

Section 1: Environment Setup & Data Preparation (15 points)

Your Tasks:

Run all setup cells successfully

Load and explore the Olivetti faces dataset

Complete Reflection Question 1 about dataset challenges

Properly split data into train/validation/test sets

Key Learning: Understanding proper data preparation and the importance of data splitting.

Section 2: Feature Extraction Deep Dive (25 points)

Your Tasks:

CODING: Extract HOG features from training data

CODING: Extract LBP features from training data

CODING: Extract features for validation and test sets

Complete Reflection Questions 2 & 3 about feature types

Compare feature dimensions and characteristics

Key Learning: How to transform raw pixels into meaningful features for machine learning.

Section 3: Classical ML Algorithms Implementation (25 points)

Your Tasks:

CODING: Train SVM with HOG and LBP features

CODING: Train Random Forest with HOG and LBP features

Complete Reflection Question 4 comparing algorithms

Analyze training vs. validation performance patterns

Key Learning: Understanding different ML algorithms and their performance characteristics.

Section 4: The Training Trap - Overfitting Demonstration (25 points)

🚨 MOST IMPORTANT SECTION Your Tasks:

Observe the "perfect" 99% accuracy model creation

Witness the reality check when testing on new data

Complete Reflection Questions 5 & 6 about overfitting

Compare overfitted vs. reasonable model approaches

Key Learning: Why 99% training accuracy often means failure and how to build models that work in production.

Section 5: Final Model Selection and Evaluation (10 points)

Your Tasks:

CODING: Select the best model based on validation performance

Complete Final Reflection Question 7 with comprehensive analysis

Justify your model selection decision

Key Learning: How to make data-driven decisions about model selection.

🎯 Coding Sections - What You Need to Complete

Throughout the notebook, you'll find "STUDENT CODING SECTION" areas marked with:

Python

YOUR CODE HERE variable_name = None # Replace None with your code # END YOUR CODE HERE

You must complete ALL coding sections for full credit.

Specific Coding Tasks:

1.Feature Extraction: Call functions to extract HOG and LBP features

2.Algorithm Training: Train SVM and Random Forest with different features

3.Model Selection: Choose the best model based on performance metrics

💡 Hints are provided in each section to guide you!

📝 Reflection Questions - Critical Thinking Required

You must answer ALL 7 reflection questions thoughtfully. These are not just "run the code" questions - they require you to think deeply about the concepts and demonstrate understanding.

Question Topics:

1.Dataset challenges and overfitting potential

2.HOG feature analysis and visualization

3.Feature comparison and trade-offs

4.Algorithm performance comparison

5.Critical: Understanding the overfitting disaster

6.Critical: Comparing overfitted vs. reasonable models

7.Final: Comprehensive model selection analysis

⚠️ One-sentence answers will not receive full credit. Show your understanding!

📊 Grading Rubric (100 points total)

Component	Points	Criteria
Code Completion	40	All "YOUR CODE HERE" sections completed correctly
Code Execution	20	Notebook runs without errors, produces expected outputs
Reflection Questions	30	Thoughtful, detailed answers showing deep understanding
File Naming	5	Correct naming convention followed
Overall Quality	5	Professional presentation, clear outputs

Detailed Grading Criteria:

Code Completion (40 points):

•Feature extraction functions called correctly (15 points)

•Algorithm training completed (15 points)

•Model selection implemented (10 points)

Reflection Questions (30 points):

•Questions 1-4: 4 points each (16 points total)

•Questions 5-6 (Overfitting): 5 points each (10 points total)

•Question 7 (Final analysis): 4 points

Quality Indicators:

• ✅ All cells execute without errors

• ✅ Visualizations display correctly

• ✅ Reflection answers demonstrate understanding beyond surface level

• ✅ Professional presentation and organization

🚨 Critical Success Factors

To Earn Full Credit:

1.Complete ALL coding sections - Don't leave any None values

2.Answer reflection questions thoughtfully - Show your understanding

3.Focus on the overfitting concepts - This is the most important learning

4.Follow naming convention exactly - Automatic 10-point deduction if incorrect

5.Test your notebook - Make sure it runs completely before submission

Common Mistakes to Avoid:

- ❌ Skipping reflection questions or giving superficial answers
- ❌ Not completing coding sections (leaving None values)
- ❌ Incorrect file naming
- ❌ Submitting a notebook that doesn't run
- ❌ Missing the key insights about overfitting

📅 Submission Instructions

What to Submit:

1.Your completed Jupyter notebook with the correct naming convention

2.All code cells executed with visible outputs

3.All reflection questions answered in the markdown cells

How to Submit:

1.Download your completed notebook from your chosen environment

2.Rename the file using the exact naming convention

3.Upload to Canvas in the Lab 03 assignment area

4.Verify your submission - Make sure the file uploaded correctly

Submission Checklist:

All coding sections completed

All reflection questions answered thoughtfully

Notebook runs completely without errors

File named correctly: L03_[LastName][FirstName]_ITAI_1378.ipynb

Submitted to correct Canvas assignment

🆘 Getting Help

If You're Stuck:

1.Read the hints provided in each coding section

2.Review the explanations in the markdown cells

3.Check the function definitions - they show you what to call

4.Ask questions in the discussion forum - Help your classmates too!

5.Attend office hours for personalized assistance

Technical Issues:

•Environment problems: Try a different platform (Colab, Kaggle, etc.)

•Library errors: Use the cloud environments which have everything pre-installed

•Code errors: Check the validation messages in each section

🎓 Learning Beyond the Lab

This lab teaches you fundamental concepts that apply to all machine learning:

•Proper data splitting prevents overfitting

•Feature engineering is crucial for classical ML

•Validation performance matters more than training performance

•Real-world deployment requires models that generalize

These concepts will serve you throughout your AI/ML career!

📞 Contact Information

•Questions about content: Post in Canvas discussion forum

•Technical issues: Email instructor with specific error messages

•Submission problems: Contact Canvas support or instructor

Good luck with your first computer vision project! Remember: the goal isn't perfect training accuracy - it's building models that work in the real world! 🌍