# **Assignment 3 (10%)**

**Real-Time Image Classification with CNNs Using Transfer Learning (ResNet)**

DUE DATE: Friday, December 6th, 2024, 11:59 PM ET

**Assignment Guidelines:**

* All students must adhere to the [Student Academic Integrity](https://intranet.laurentian.ca/policies/2017.09.19%20-%20Policy%20and%20Procedures%20on%20Academic%20Integrity%20-%20EN.pdf).
* Assignments must follow the programming standards outlined in the document on the course website via D2L. Marks will be deducted if these standards are not followed.
* **Submit the python** .py **or** .ipynb **files and an associated PDF report per group**. Name the file according to your group "ID" and assignment number, per this example for Assignment 1 and Group 1: COSC\_4117EL\_A3\_G1-\*.py. Apply the same naming convention for the PDF: COSC\_4117EL\_A3\_G1.pdf.
* **Do NOT zip the files** when submitting.
* Multiple submissions are allowed, but only the most recent version will be marked.
* A late penalty of 2% per hour (or part thereof) will be applied after the deadline. After 49 hours, the penalty will be 100%, and submissions will no longer be accepted.
* These assignments are key learning tools for the exams. Code your assignments independently within your group. Plagiarism detection software may be used to compare all submissions, and academic dishonesty will be pursued vigorously.
* **Generative AI tools (e.g. ChatGPT, Gemini, Autopilot, etc.) are permitted,** provided their use is properly cited. Ensure that any text or code generated through these tools is acknowledged in your code comments and report. **Failure to cite generative AI usage will be considered a violation of academic integrity.**

**Introduction:**

In this assignment, you will explore the power of Convolutional Neural Networks (CNNs) paired with transfer learning to tackle a real-world problem—real-time image classification from a live webcam feed. Using a pre-trained [ResNet models](https://pytorch.org/vision/main/models/resnet.html), your goal is to **design and develop a system capable of accurately classifying images in real-time**. You will document your process and results in a comprehensive report and present the functionality of your model through a video demonstration. Please upload the video to YouTube and include the link in your report.

**PART 1: Problem Proposal and Justification (in your report) [10 marks]**

1. Identify a Problem::
   1. Choose a **creative real-time classification problem/application** that can be addressed using a webcam. Examples might include recognizing gestures, facial expressions, specific objects, or particular activities.
   2. **Justify the significance of your chosen problem**, explaining its relevance and potential real-world impact.
2. Dataset Description:
   1. **Describe the dataset required to train your model**. Outline the classes, diversity, and how you plan to gather or access this data.
   2. **Explain any data augmentation techniques you use** to enhance the robustness and generalization of your model.

P**ART 2: Implementation Guide (in your report) [20 marks]**

1. Transfer Learning Approach:
   1. Explain your choice of a pre-trained ResNet model for this task. **Specify which ResNet version you would use and why it is suitable**.
   2. **Detail any modifications made** to the network architecture to align it with your specific classification problem.
2. Real-Time Processing:
   1. Outline the steps needed to process images from a webcam for real-time classification.
   2. Describe any custom adaptations (if applicable) to ResNet required for real-time classification.

***Utilizing Libraries and Tools:***

* *You are encouraged to use libraries like PyTorch for CNN and ResNet, and OpenCV for image processing tasks.*
* *Use the sample code provided as a base, tailoring it to meet the specific needs of your project.*

P**ART 3: Performance Evaluation and Testing (in your report) [10 marks]**

1. Evaluation Metrics:
   1. Define the metrics (e.g., accuracy, precision, recall, F1 score) used to evaluate your model’s performance, and explain why these metrics are relevant to your problem.

2. Discuss anticipated challenges in real-time processing and potential solutions.

**Video Demonstration:**

* Record a video showcasing the testing of your CNN model. In the video, explain the model’s features, the real-time classification process, and observed results.
* Upload the video to YouTube and include the link in your report as a visual and practical complement to your documentation.

**Submission:**

1. **Code**: Submit all implementation files (.py) as necessary, ensuring each file is clearly named.
2. **Report**: Submit a detailed PDF report explaining your approach, challenges, and key decisions. Include the **YouTube video link**, visualizations, tables, or charts supporting your findings.
3. Ensure your assignment is self-contained, with all relevant information in the report and code for clarity and completeness.

**Group Contributions and Grading:**

If any group member feels another member’s contribution is not proportional, this can be addressed. Include an additional section in the report detailing the situation and the proposed grade adjustment, with the agreement of all group members. Ensure all group members provide consent to any grade changes.

Open communication and collaboration are always encouraged. However, if discrepancies in contributions are significant, this mechanism ensures fairness in grading.

**Best of luck! Remember, the learning process is just as important as the final result.**