# Final Project Instructions – Deep Learning Fall 2025

# 521153S-3006-2025

## Transfer Learning for Multi-label Medical Image Classification

### Motivation

In medical imaging, obtaining large-scale, labeled datasets is often challenging due to privacy concerns, high annotation costs, and limited availability of expert knowledge. To effectively learn and boost performance on small-scale datasets, we leverage transfer learning techniques which consist of models that are trained on large amounts of data.

### Goal

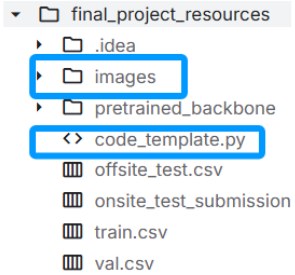
Improve the performance of multi-label retinal image classification using transfer learning by fine-tuning models, while deepening the understanding of deep learning techniques.

### Task Overview

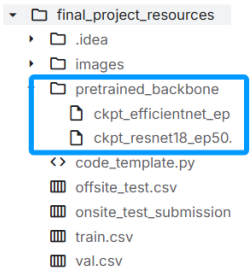
In this project, we address the problem of multi-label retinal disease detection, focusing on three major conditions: **Diabetic Retinopathy (DR), Glaucoma (G), and Age-related Macular Degeneration (AMD)**. To tackle the challenge of limited annotated medical data, we adopt transfer learning strategies, leveraging models pretrained on large-scale datasets and fine-tuning them for multi-label retinal image classification. The experiments are conducted on the **ODIR dataset**[1]**,** which is divided into a training set of 800 images, a validation set of 200 images, an offsite test set of 300 images, and an onsite test set of 250 images, with all images standardized to a resolution of 256×256. The evaluation metrics include **precision, recall, F-score** of each disease and the **average F-score** over the three diseases.

### Task 1: Transfer Learning (10 points)

1) Download the ODIR dataset from the https://www.kaggle.com/t/a56eadae07e9477d9b7f3ee5f85dffc5 along with the template code:



2) We provide two pretrained models (**EfficientNet**[2] **and ResNet18**[3]) that have been trained on three large-scale datasets: ADAM[4], REFUGE2[5], and APTOS[6]. You can download them from the same link as above:




3) You task is to perform transfer learning with three different setups using EfficientNet and ResNet18 and evaluate their performances on both offsite test set and onsite test set:
   - **Task1.1: No fine-tuning**: Evaluate directly on ODIR test set. (2 points)
   - **Task1.2: Frozen backbone, fine-tune classifier only**: Backbone weights are fixed, classifier is updated on ODIR training set. (4 points)
   - **Task1.3: Full fine-tuning**: Both backbone and classifier are updated on ODIR training set. (4 points)
4) Grading criteria:
   - Comparative to the reference performances on **onsite test set**: 100%*full points of the task
   - Lower: 50%*full points of the task

*Please report all the evaluation metrics on the offsite test set in the technique report. Provide an in-depth analysis, for example by discussing the advantages and disadvantages of the model. For the onsite test set, you only need to submit your predictions to the Kaggle competition to obtain the average F-score, which will be used to evaluate the model's performance, and then report it in your technique report. An in-depth analysis is not possible, since the labels of the onsite test set are not accessible.*

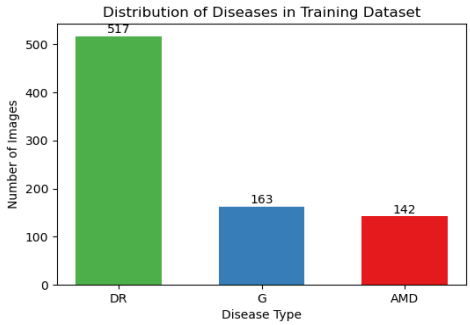
We provide the reference average F-score on onsite test set for Task 1 in Table 1.

Table 1: Reference performances on onsite test set for task 1.

| methods | No fine-tuning | fine-tune classifier only | Full fine-tuning |
|---|---|---|---|
| EfficientNet | 60.4 | $73.5_{\pm 0.6}$ | $80.4_{\pm 0.5}$ |
| ResNet18 | 56.7 | $61.4_{\pm 0.3}$ | $78.8_{\pm 0.8}$ |

**Task 2: Loss Function (10 points)**

The distribution of different diseases in the training set is as follows:




It can be easily observed that the number of DR cases is significantly higher than that of G and AMD. This may lead the model to overlook the learning of minor classes. The objective of this task is to explore the use of Focal Loss[7] and Class-Balanced Loss to address the problem of class imbalance. The requirements are as follows:

1) Implement two loss functions:
   - **Task2.1** Focal Loss: A loss function designed to address class imbalance by down-weighting easy examples and focusing training on hard, misclassified ones. (5 points)
   - **Task 2.2** Class-Balanced Loss: Re-weight the BCE loss according to class frequency. This is a common method for handling class imbalance. (5 points)
2) Evaluate the impact of loss function on model performance.
3) Grading criteria:
   - Comparative to the reference performances by full fine-tuning on **onsite test set**: 70%*full points of the task
   - Lower: 50%*full points of the task
   - Better than the reference performances by 0.5%: 85%*full points of the task
   - Better than the reference performances by 1.0%: 100%*full points of the task

*You only need to select the better-performing model between ResNet18 and EfficientNet for the report.*

**Task 3: Attention Mechanisms (15 points)**

1) Implement two attention mechanisms:
   - **Task3.1** Squeeze-and-Excitation (SE)[8] (6 points)
   - **Task3.2** Multi-head Attention (MHA)[9][9] (9 points)
2) Evaluate the impact of attention mechanism on model performance.
3) Grading criteria:
   - Comparative to the reference performances by full fine-tuning on **onsite test set**: 70%*full points of the task
   - Lower: 50%*full points of the task

- Better than the reference performances by 1.0%: 85%*full points of the task
- Better than the reference performances by 1.5%: 100%*full points of the task

*You only need to select the better-performing model between ResNet18 and EfficientNet for the report.*

**Task 4: Open Questions (10 points)**

In this task, you are encouraged to explore and incorporate any additional modules or effective learning strategies to further improve the predictive performance of the models. Below are some possible directions you may consider:

1) Fine-tune more powerful backbone networks such as Swin Transformer[10] and Vision Transformer[11] to improve the disease detection performances.
2) Use Explainable AI techniques such as **GradCAM**[12] to analyze what features in the image are contributing the most and the least in the model's decision-making process, , then use the attention map to guide the learning, thereby improving the performances.
3) Try out the different ensemble learning methods (Stacking, Boosting, Weighted Average, Max Voting, Bagging) and analyze whether the performance increases or not.
4) Use a simple image generation model such as a VAE[13] to generate new retinal images in order to augment the training set, and then analyze whether the augmented dataset leads to improvements in model performance.
5) Grading criteria:
    - Comparative to the reference performances by full fine-tuning on **onsite test set**: 70%*full points of the task
    - Lower: 50%*full points of the task
    - Better than the reference performances by 1.0%: 85%*full points of the task
    - Better than the reference performances by 1.5%: 100%*full points of the task

*You only need to select the better-performing model between ResNet18 and EfficientNet for the report.*

**Task 5: Technique Report (15 points)**

The final technique report must not exceed three pages and include the key sections: Introduction, Methods, Results, and Discussions, as well as all relevant figures and tables. The criteria for technique report are below:

- The report should be complete, clearly structured, and concise.
- Minor typos are acceptable, but overall readability and clarity are required.
- In-depth experimental analysis including but not limited to reporting the precision, recall, and F-score of each disease on the offset test set, detailed explanation, advantages and disadvantages of the techniques used each task etc.

*PS: You may use large language models such as ChatGPT for language polishing, but direct*

*report generation is strictly prohibited.*

**Making your online submission:**

**The results of the onsite test set for task 1-4 must be included in the report.**

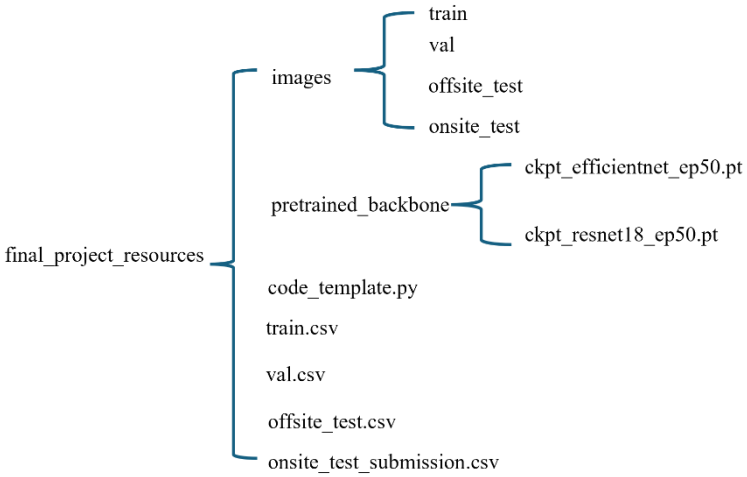
To make your submission and get the performance on onsite test set, please follow the template code and submit it to Kaggle for online evaluation. The link to the Kaggle competition is: https://www.kaggle.com/t/a56eadae07e9477d9b7f3ee5f85dffc5. The task here is to predict labels of three diseases for those images and then submit those outputs as a csv file in the following form:

| id | D | G | A |
|---|---|---|---|
| 4595_right.jpg | 0 | 0 | 0 |
| 4155_left.jpg | 0 | 0 | 0 |
| 597_left.jpg | 0 | 0 | 0 |
| 4268_right.jpg | 0 | 0 | 0 |

In Kaggle, we provide a submission template: onsite_test_submission.csv.

**Resources Checklist:**

For completing this project, the following resources will be provided:




You can down these from https://www.kaggle.com/t/a56eadae07e9477d9b7f3ee5f85dffc5.

**Report Template:**

The following is the download link for the technique report template:

Word: WordTemplate.doc

Latex: ICIP LaTeX Template - Overleaf, Online LaTeX Editor

**Contribution Statement**

The final project is a **team assignment**. You are free to form your own groups, with each group consisting of no more than three members. In the technique report, each team should provide a detailed description of the division of work among members, specifying which tasks each member carried out and which parts of the report each member was responsible for writing. In principle, if the contributions of the members are comparable, all members of the same team will receive the same score; however, if a member's contribution is significantly smaller, their score will be reduced accordingly.

**Deliverables Checklist:**

We will randomly check whether each group's experimental results match those reported in the report. So, please provide the following deliverables:

- technique report
- source code
- the trained model

*1）Students can choose to complete the final project individually or in a group (Please fill out the Google form https://docs.google.com/spreadsheets/d/1f1q8Z3qPDweXrKNAVQLySbeU_j4c9MR1-ibUv2NDlt8/edit?usp=sharing). Each group can have up to three students (Each student's contribution needs to be stated in the report). Please include your team name(s) of Kaggle in the Google form above.*

*2）Please submit the zip file, which includes your code, trained model, and a readme.md file (explaining how to run the code). Please name the zip file, source code, trained models according to the following rule: teamname.zip, teamname.py, and teamname_task#.pt. For example, if your team's name is **sam**, the name of the zip file should be sam.zip, and the name of the source code should be sam.py. For pretrained model, for example, for the second task in Task 1 (fine-tuning the classifier), the filename should be sam_task1-2.pt. For the second task in Task 2 (Class-Balanced Loss), the filename should be sam_task2-2.pt.*

**References**

[1] Li N, Li T, Hu C, et al. A benchmark of ocular disease intelligent recognition: One shot for multi-disease detection. International symposium on benchmarking, measuring and optimization. Cham: Springer International Publishing, 2020: 177-193.

[2] Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks. International conference on machine learning. PMLR, 2019: 6105-6114.

[3] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[4] Fang H, Li F, Fu H, et al. Adam challenge: Detecting age-related macular degeneration from fundus images. IEEE transactions on medical imaging, 2022, 41(10): 2828-2847.

[5] Fang H, Li F, Wu J, et al. Refuge2 challenge: A treasure trove for multi-dimension analysis and evaluation in glaucoma screening. arXiv preprint arXiv:2202.08994, 2022.

[6] Karthik, Maggie, and Sohier Dane. APTOS 2019 Blindness Detection. https://kaggle.com/competitions /aptos2019-blindness-detection, 2019. Kaggle.

[7] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection. Proceedings of the IEEE international conference on computer vision. 2017: 2980-2988.

[8] Hu J, Shen L, Sun G. Squeeze-and-excitation networks. Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.

[9] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. Advances in neural information processing systems, 2017, 30.

[10] Liu Z, Lin Y, Cao Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows. Proceedings of the IEEE/CVF international conference on computer vision. 2021: 10012-10022.

[11] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020.

[12] Selvaraju R R, Cogswell M, Das A, et al. Grad-cam: Visual explanations from deep networks via gradient-based localization. Proceedings of the IEEE international conference on computer vision. 2017: 618-626.

[13] Kingma D P, Welling M. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114, 2013.

## Key Terminologies

- **Fine-tuning:** It is the process of taking a pre-trained model and further training it on a specific dataset to boost performance. This may include unfreezing one or more layers of the pretrained model.
- **ADAM:** An open-access dataset for age-related macular degeneration (AMD) detection, as well as lesion and structure segmentation.
- **REFUGE2:** An open-access dataset for glaucoma detection and optic disc/cup segmentation.
- **APTOS:** An open-access dataset for diabetic retinopathy grading.
- **Kaggle:** A community platform for data science and machine learning competitions and datasets.
- **Attention:** Attention mechanisms that allow the model to focus on specific regions of an input image that are most relevant to the task at hand.
- **GradCAM:** Gradient-weighted Class Activation Mapping (Grad-CAM) is a technique used to visualize which parts of an image contribute the most to a model's prediction.
- **Ensemble Learning:** This involves combining multiple models to improve predictive performance. Ensemble methods can help reduce overfitting and increase model robustness.
- **VAE:** Variational Autoencoder (VAE) is a generative model that learns a compressed latent representation of data and can generate new samples by decoding from this latent space.
- **Recall:** Recall measures how many of the actual positives were correctly identified. It is calculated as:

$$Recall = \frac{TP}{TP + FN}$$

TP: The model predicts positive, and the actual label is positive.
FN: The model predicts negative, but the actual label is positive.

- **Precision:** Precision measures how many of the predicted positives are correct. It is calculated as:

$$Precision = \frac{TP}{TP + FP}$$

TP: The model predicts positive, and the actual label is positive.
FP: The model predicts positive, but the actual label is negative.

- **F-score:** The F-score is the harmonic mean of precision and recall. It is calculated as:

$$F = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

What is Fine-tuning: https://www.techtarget.com/searchenterpriseai/definition/fine-tuning

Kaggle: https://www.kaggle.com/

ADAM dataset: Home - iChallenge-AMD - Grand Challenge

REFUGE2 dataset: Details - REFUGE - Grand Challenge

APTOS dataset: APTOS 2019 Blindness Detection | Kaggle

GradCAM Article: https://datascientest.com/en/what-is-the-grad-cam-method

Ensemble Learning and techniques: https://www.geeksforgeeks.org/a-comprehensive-guide-to-ensemble-learning/

VAE: Variational autoencoder - Wikipedia