National College of Ireland

Configuration Manual

MSc Research Project
Master of Science in AI

Yixuan Wang
Student ID: 23162767

School of Computing
National College of Ireland

Supervisor: Dr. Paul Stynes

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: ……Yixuan Wang……

Student ID: ……23162767……

Programme: ……MSc in AI…… **Year:** …Jan…2024……

Module: …… MSc Practicum Part 2……

Lecturer: …… Professor Paul Stynes……

Submission Due Date: …… 28th January 2025……

Project Title: … A Real-Time Human Action Recognition Model for Assisted Living…

Word Count: ……1067…… **Page Count:** ……7……

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: …… Yixuan Wang ……

Date: ……11th December ……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	☑
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☑
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☑

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Yixuan Wang
Student ID: 23162767

1 Preparation of MMAction2 Framework

The setup of the MMAction2 framework[1], an open-source ecosystem based on PyTorch with specific tested dependencies, is described below.

```
conda deactivate
conda create --name openmmlab_zz python=3.8 -y
conda activate openmmlab_zz
conda install pytorch==2.1.2 torchvision==0.16.2 torchaudio==2.1.2 pytorch-cuda=11.8 -c
pytorch -c nvidia
pip install fsspec
pip install -U openmim
mim install mmengine
mim install mmcv==2.1.0
mim install mmdet==3.2.0  # optional
mim install mmpose  # optional
git clone https://github.com/open-mmlab/mmaction2.git
cd mmaction2
pip install -v -e .
```

Test the framework:

```
mim download mmaction2 --config tsn_imagenet-pretrained-r50_8xb32-1x1x8-
100e_kinetics400-rgb --dest .
python demo/demo.py tsn_imagenet-pretrained-r50_8xb32-1x1x8-100e_kinetics400-
rgb.py     tsn_imagenet-pretrained-r50_8xb32-1x1x8-100e_kinetics400-rgb_20220906-
2692d16c.pth     demo/demo.mp4 tools/data/kinetics/label_map_k400.txt
```

2 Dataset Creation

This study utilised the NTU RGB+D Action Recognition Dataset[2](J. Liu et al., 2019). All RGB video samples of Falling (948 videos), Staggering (948 videos), and Chest pain (948 videos) in the "Medical Conditions" category were selected to represent dangerous scenarios. The "Normal Scenario" class was formed by randomly selecting 80 videos from each of the 40 classes in the "Daily Actions" category to include a larger sample size (3,200 videos) with a wide diversity of daily activities, which simulate real-life conditions.

---

[1] https://mmaction2.readthedocs.io/en/latest/get_started/installation.html
2 https://rose1.ntu.edu.sg/dataset/actionRecognition/

The dataset was split into training, validation and testing sets, in proportion of 75%, 12.5% and 12.5% respectively. The splitting process ensured that the proportion of the four classes remained the same in training, testing and validation sets. Then, feature-label mapping was performed. An annotation text file for each set was created, listing the relative video path with its corresponding label. The dataset creation and label mapping were performed in "**customise_datasets.ipynb**", which is located in the main folder. The created training, validation and testing datasets were saved in "**mmaction2/data/normal_vs_3critical**", including both videos and the annotation test files.

3 Code Structure

The code is organised in a modular approach. It is uploaded in the GitHub repository, and its main components are outlined below.

- "**config**" folder holds all model configuration scripts. In each config file, the inherited based model, training, validation and test pipelines, data loaders, and model training settings, are defined.
- "**mmaction2/configs/recognition/slowfast**" holds the downloaded pretrained models and customised models for SlowFast.
- "**mmaction2/configs/recognition/i3d**" contains models for I3D.
- "**mmaction2/configs/recognition/uniformerv2**" contains defined models for UniFormer.
- "**mmaction2/configs/recognition/timesformer**" holds all models for TimeSformer.
- The model training script is located at "mmaction2/tools/**train.py**", and the testing script is defined in "mmaction2/tools/**test.py**"
- All training and testing logs, including model checkpoints, are saved in "mmaction2/**work_dirs**".
- The script used for calculating metrics (except for FLOPs and Parameters) and ploting graphs is located at "mmaction2/1_Evaluation_Plots_and_Investigation/**evaluation_metric_analysis.ipynb**". Plots and analysis results presented in the report are located at the same folder "mmaction2/**1_Evaluation_Plots_and_Investigation**"
- Parameter size and Flops were calculated by calling "mmaction2/tools/analysis_tools/**get_flops.py**"
- The script for loading model checkpoints and making inference for entire test data is located at "mmaction2/ **Inference_test_data.py**"
- The script for make inference based on a model config file, a checkpoint file and a video path is located at "mmaction2/**Inference.py**". The same code for calling using command-line arguments is defined at "mmaction2/**Inference2.py**"

4 Commands to run training, testing and evaluation files

Multiple models were trained and evaluated. The commands to train, test and evaluate the models are listed below.

All code should run in the conda environment, which should be activated as:
"conda deactivate
conda activate openmmlab_z"

4. 1 Train models

4.1.1 SlowFast:

python tools/train.py configs/recognition/slowfast/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50.py

python tools/train.py configs/recognition/slowfast/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_epoch70 lr1e-5.py

python tools/train.py configs/recognition/slowfast/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_epoch70_lr1e-4.py

Train with backbone 101:
python tools/train.py configs/recognition/slowfast/slowfast_r101_8xb8-8x8x1-256e_kinetics400-rgb_transfer.py

4.1.2 I3D:

python tools/train.py configs/recognition/i3d/i3d_imagenet-pretrained-r50_8xb8-32x2x1-100e_kinetics400-rgb_transfer_epoch60.py

4.1.3 TimesFormer:

python tools/train.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST.py

python tools/train.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_dividedST.py

4.1.4 Uniformerv2:

python tools/train.py configs/recognition/uniformerv2/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer.py

4.2 Test model and save the results:

Different checkpoint paths with the best top-1 accuracy or the last epoch were selected to test model performance on the test set:

4.2.1 SlowFast

python3 tools/test.py **configs/recognition/slowfast/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50.py** work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50/best_acc_top1_epoch_50.pth --work-dir work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50/test_results_last

python3 tools/test.py **configs/recognition/slowfast/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50.py** work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50/best_acc_top1_epoch_50.pth --work-dir work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50/test_results_best2

Test these checkpoints:

mmaction2/work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_epoch70_lr1e-5/last_checkpoint

mmaction2/work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50/best_acc_top1_epoch_50.pth

Test r101 backbone model:
python3 tools/test.py configs/recognition/slowfast/slowfast_r101_8xb8-8x8x1-256e_kinetics400-rgb_transfer.py work_dirs/slowfast_r101_8xb8-8x8x1-256e_kinetics400-rgb_transfer/best_acc_top1_epoch_50.pth --work-dir work_dirs/slowfast_r101_8xb8-8x8x1-256e_kinetics400-rgb_transfer/tests_last

4.2.2 I3D:

Using the best top-1 accuracy checkpoint:
python3 tools/test.py **configs/recognition/i3d/i3d_imagenet-pretrained-r50_8xb8-32x2x1-100e_kinetics400-rgb_transfer_epoch60.py** work_dirs/i3d_imagenet-pretrained-r50_8xb8-32x2x1-100e_kinetics400-rgb_transfer_epoch60/best_acc_top1_epoch_56.pth --work-dir work_dirs/i3d_imagenet-pretrained-r50_8xb8-32x2x1-100e_kinetics400-rgb_transfer_epoch60/test_results_best2

Using the last epoch checkpoint:
python3 tools/test.py **configs/recognition/i3d/i3d_imagenet-pretrained-r50_8xb8-32x2x1-100e_kinetics400-rgb_transfer_epoch60.py** work_dirs/i3d_imagenet-pretrained-r50_8xb8-32x2x1-100e_kinetics400-rgb_transfer_epoch60/epoch_60.pth --work-dir work_dirs/i3d_imagenet-pretrained-r50_8xb8-32x2x1-100e_kinetics400-rgb_transfer_epoch60/test_results_last

4.2.3 TimeSFormer

Using the best top-1 accuracy checkpoint:
python3 tools/test.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST.py work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/best_acc_top1_epoch_30.pth --work-dir work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/test_results_best2

python3 tools/test.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST.py work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/epoch_33.pth --work-dir work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/test_results_last

python3 tools/test.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST.py work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/best_acc_top1_epoch_42.pth --work-dir work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/second_run_tests_best_top1

python3 tools/test.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST.py work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/best_mean_acc_epoch_48.pth --work-dir

work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/second_run_tests_best_mean

python3 tools/test.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST.py work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/epoch_50.pth --work-dir work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/second_run_tests_last

Test divided version:
python3 tools/test.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_dividedST.py work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_dividedST/best_acc_top1_epoch_40.pth --work-dir work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_dividedST/tests_best_top1

Test using last checkpoint:
python3 tools/test.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_dividedST.py work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_dividedST/epoch_50.pth --work-dir work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_dividedST/tests_last

4.2.4 UniformerV2

python3 tools/test.py **configs/recognition/uniformerv2/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer.py** work_dirs/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer/best_acc_top1_epoch_38.pth --work-dir work_dirs/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer/test_results_best2

python3 tools/test.py **configs/recognition/uniformerv2/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer.py** work_dirs/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer/resume_50_checkpoints/epoch_10.pth --work-dir work_dirs/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer/test_resume_50epochs_best

python3 tools/test.py **configs/recognition/uniformerv2/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer.py** work_dirs/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer/resume_50_checkpoints/epoch_12.pth --work-dir work_dirs/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer/tests_results_resume_50epochs_last

4.3 Plot evaluation accuracy vs iterations

4.3.1 Slow Fast

Accuracy vs Iterations

python tools/analysis_tools/analyze_logs.py plot_curve "work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-

rgb_transfer_2_NorVsCrit_epoch50/20241123_085206/vis_data/20241123_085206.json" --title "SlowFast Validation Accuracy vs. Iterations" --out "SlowFast_accuracy vs. Iterations.png"

Loss vs iterations
python tools/analysis_tools/analyze_logs.py plot_curve "work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50/20241123_085206/vis_data/20241123_085206.json" --title "SlowFast Validation Loss vs. Iterations" --out "SlowFast_loss vs. Iterations.png" --keys loss

python tools/analysis_tools/analyze_logs.py plot_curve work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_epoch70_lr1e-5/20241126_111502/vis_data/20241126_111502.json --title "SlowFast (LR=1e-5, Epochs=70) Validation Loss vs. Iterations" --out "SlowFast_epoch70_lr1e-5_loss vs. Iterations.png" --keys loss

python tools/analysis_tools/analyze_logs.py plot_curve work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_epoch50_lr1e-4/20241126_140222/vis_data/20241126_140222.json --title "SlowFast (LR=1e-4, Epochs=50) Validation Loss vs. Iterations" --out "SlowFast_epoch50_lr1e-4_loss vs. Iterations.png" --keys loss

4.3.2 I3D
Accuracy vs Iterations
python tools/analysis_tools/analyze_logs.py plot_curve "work_dirs/i3d_imagenet-pretrained-r50_8xb8-32x2x1-100e_kinetics400-rgb_transfer_epoch60/20241126_031630/vis_data/20241126_031630.json" --title "Inflated 3D ConvNet Validation Accuracy vs. Iterations" --out "I3D_accuract vs. Iterations.png"

Loss vs iterations
python tools/analysis_tools/analyze_logs.py plot_curve "work_dirs/i3d_imagenet-pretrained-r50_8xb8-32x2x1-100e_kinetics400-rgb_transfer_epoch60/20241126_031630/vis_data/20241126_031630.json" --title "Inflated 3D ConvNet Validation Loss vs. Iterations" --out "I3D_loss vs. Iterations.png" --keys loss

4.3.3 TimesFormer
Loss vs iterations
python tools/analysis_tools/analyze_logs.py plot_curve "work_dirs/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST/20241127_120043/vis_data/20241127_120043.json" --title "TimeSformer Validation Loss vs. Iterations" --out "TimeSFormer_loss vs. Iterations.png" --keys loss

4.3.4 UniFormerv2
python tools/analysis_tools/analyze_logs.py plot_curve "work_dirs/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer/20241128_011657/vis_data/20241128_011657.json" --title "Uniformerv2 Validation Loss vs. Iterations" --out "UniformerV2_loss vs. Iterations.png" --keys loss

4.4 Calculate Flops, Parameter Size

4.4.1 SlowFast

python tools/analysis_tools/get_flops.py configs/recognition/slowfast/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50.py --shape 1 3 32 224 224

slowfast r101:
python tools/analysis_tools/get_flops.py configs/recognition/slowfast/slowfast_r101_8xb8-8x8x1-256e_kinetics400-rgb_transfer.py --shape 1 3 32 224 224

4.4.2 I3D

python tools/analysis_tools/get_flops.py configs/recognition/i3d/i3d_imagenet-pretrained-r50_8xb8-32x2x1-100e_kinetics400-rgb_transfer_epoch60.py --shape 1 3 32 224 224

4.4.3 TimeSformer

python tools/analysis_tools/get_flops.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_jointST.py --shape 1 3 8 224 224

For divided version:
python tools/analysis_tools/get_flops.py configs/recognition/timesformer/timesformer_spaceOnly_8xb8-8x32x1-15e_kinetics400-rgb-transfer_dividedST.py --shape 1 3 8 224 224

4.4.4 UniFormer

python tools/analysis_tools/get_flops.py configs/recognition/uniformerv2/uniformerv2-base-p16-res224_clip_8xb32-u8_kinetics700-rgb_transfer.py --shape 1 3 8 224 224

4.5 Inference:

Inference2.py is created to make inference by passing in a config file, checkpoint path and a video path as command-line arguments.
Example code:
python Inference2.py **configs/recognition/slowfast/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50.py** work_dirs/slowfast_r50_8xb8-4x16x1-256e_kinetics400-rgb_transfer_2_NorVsCrit_epoch50/epoch_50.pth data/normal_vs_3critical/test/S001C001P007R001A045.mp4

5 Server Specifications

Memory	64GB Ram
CPU	AMD Ryzen™ 7 5800X Processor
Storage	2x (Samsung 990 PRO NVMe M.2 SSD, 2 TB)
GPU	Tesla V100 GPU