國立臺灣科技大學資訊工程系

112 學年度第 1 學期專題研究

總報告

基於面部情緒辨識技術的實況直播情緒共鳴應用

研究組員

B10915065 李昶勳

B10915036 余修辰

B10915059 許鎮承

B10915060 楊登傑

指導教授：戴碧如

中　華　民　國　　年　　月　　日

摘要

本專題旨在讓觀眾之間能產生共鳴，藉此提高觀賞體驗，而實況主也能依據觀眾的反應去調整實況內容。為了能即時且有效地捕捉觀眾面部情緒，在方法上，我們基於動態面部情緒辨識的模型上進行改良，透過結合微表情識別模型，以找出性能表現最佳的模型。

初步結果顯示模型能夠即時有效地偵測出觀眾的面部情緒，接著我們透過 Google Sheets API 將所有觀眾的情緒資訊進行彙整，並對這些資訊進行處理，來讓結果變的淺顯易懂，最終讓觀眾可以實時地看到其他人的情緒反應，增加觀賞體驗。

目錄

一、緒論

隨著現代生活節奏不斷加快，人們經常承受著許多壓力，為了紓解這些壓力，一些人會選擇觀賞實況直播。然而，觀賞實況直播與親身參與現場表演活動相比，觀賞實況直播缺少了現場氛圍。

假如觀眾能夠透過網際網路來傳達他們的真實情緒，引起觀眾之間的共鳴，也許能夠使觀看的效果大大提升。對於實況主方面，也能藉由觀眾的情緒反應，知道如何改進實況內容，進而提供更好的實況內容。

首先，為了研究此作品是否能夠帶來更好的影響，我們模擬了實際情境，運用面部情緒辨識技術捕捉觀眾真實的情緒反應，盡可能在短時間內將所有觀眾的情緒資訊彙整，並以易理解的方式呈現，達到共鳴的效果。

為了找出最適用於我們作品的面部情緒辨識模型，我們利用多種資料集來訓練微表情辨識模型和動態面部情緒辨識模型，比較彼此之間的優劣後，選出性能表現最佳的模型。

二、文獻回顧

1. 動態面部情緒辨識

近年來關於動態面部情緒辨識的研究愈多(如:Former-DFER[1]、IAL[2])，其跟靜態面部情緒辨識差異性在：動態基於影片，而靜態是基於圖片。另外，影片相較於圖片也比較符合真實情緒，故近年來動態面部情緒的技術備受矚目。

Former-DFER 是第一篇將 Transformer 應用在動態情緒辨識技術上的研究，在結果上也比過去常使用的 CNN 好上許多。而要辨識面部情緒時，常會面對到一個問題：有時情緒不會那麼明顯，可能會造成情緒的誤判，為了解決這個問題，IAL 提出了 loss function 去解決這個問題，不過在整體表現上還是沒有太大的提升，其一大部分原因在於資料量不足，故近期有幾篇研究採用了自監督式學習如 MAE-DFER[3]，來解決資料量不足的問題，該研究也是目前在動態面部情緒辨識資料集(FERV39K[4]、DFEW[5])表現最佳的模型，在本次的研究我們也是基於 MAE-DFER 上進行改良。

2. 宏表情情緒辨識

因為我們的作品上，需要即時且快速地了解當時觀眾的情緒，並且他們觀看時的情緒時時刻刻都在變化著，再加上每個人個體差異不同，面對同一張畫面時情緒表現不一，所以如何在一定時間內，發現他們的情緒是非常重要的。故根據 MAE-DFER 面部情緒辨識的基礎上，我們另外也尋找了基於影片的 MMEmotionRecognition[6]，和基於圖片的 POSTER-V2[7]，比較影片和圖片情緒辨識的模型之間對各種表情的辨識度。

3. 微表情情緒辨識

與上述提到的不一樣在於，上述主要關注的是宏表情，所謂的宏表情是指表情動作大於 0.5 秒的表情，而微表情則是表情動作小於 0.5 秒的表情。
因為其應用範圍廣泛，在近期也有許多人開始進行相關研究，不過在表現上還沒有太大的進展，其原因也是因為資料量不足的關係。
而在本次的研究我們是採用 MMNET[8]。

三、研究方法

3.1 資料前處理

在將資料丟到模型預測前，為了能夠準確地抓出觀眾的臉，我們使用 mediapipe 來抓取。

另外，在微表情辨識模型的輸入上，我們採用和其他研究不一樣的方式。過去研究只用到 Onset 幀和 Apex 幀，Onset 幀指的是一段時間內最一開始的幀，而 Apex 幀指的是一段時間內表情強度最大的幀。我們認為這樣會造成過多的情緒資訊被忽略，造成模型誤判。所以在這次的研究中，我們選擇將一段時間內所有的幀和 Onset 幀相減取平均，接著丟入模型進行特徵萃取。

3.2 面部情緒辨識

- 動態情緒辨識模型

本次研究中我們在一開始採用了 Former-DFER，其架構如下圖:

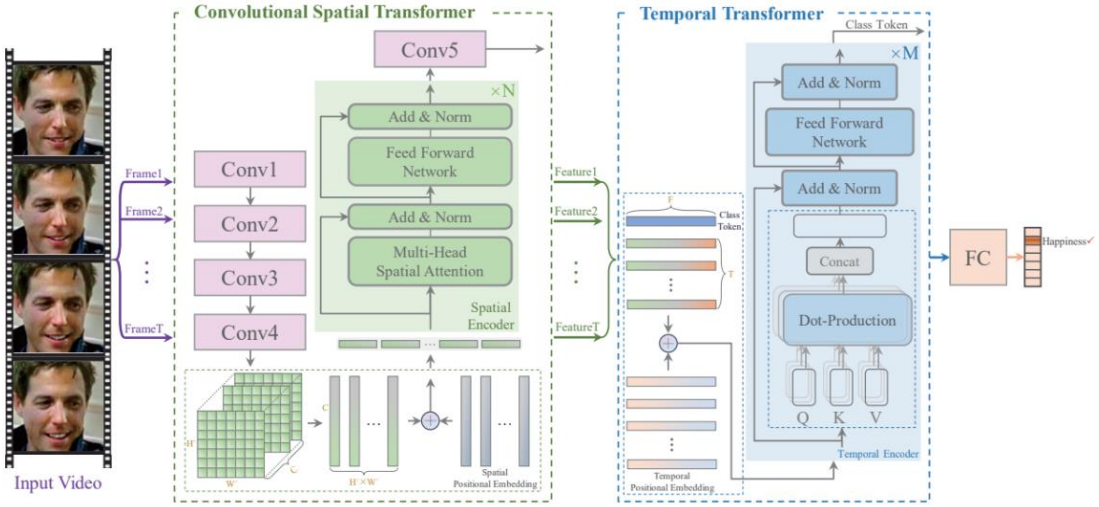

圖 1: Former-DFER 架構[1]

但在經過實驗後，發現在 Disgust 和 Fear 上表現特別差，其原因在於這兩個情緒在資料集中資料量十分稀少。

考量到 supervised(監督式學習)仰賴資料量的問題，最後我們將模型改成採用 self-supervised(自監督式學習)的模型 MAE-DFER，該模型基於 VideoMAE[9]進行改良，將 ViT 更改為 LGI-Former，以大幅減少運算成本。透過大量無標籤的資料進行預訓練，該模型在 FERV39K 的準確率為 52.07%，為目前表現最為優異的模型。LGI-Former 與模型結構如下圖:

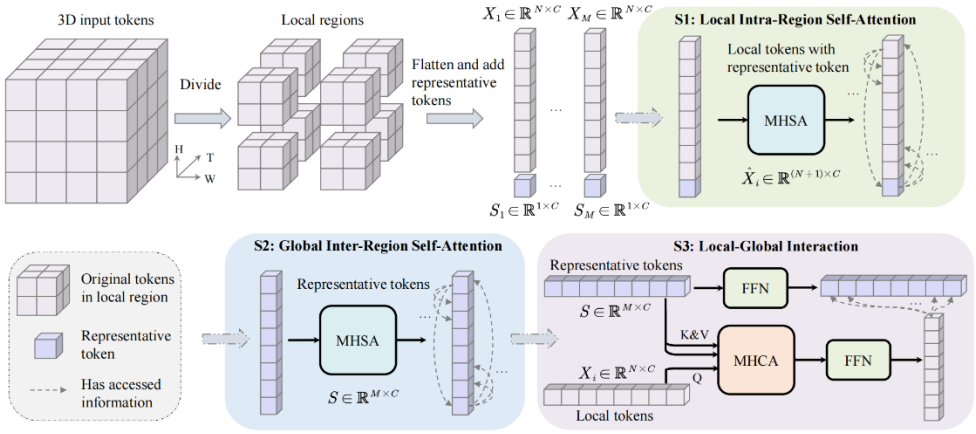

圖 2: LGI-Former 架構[3]

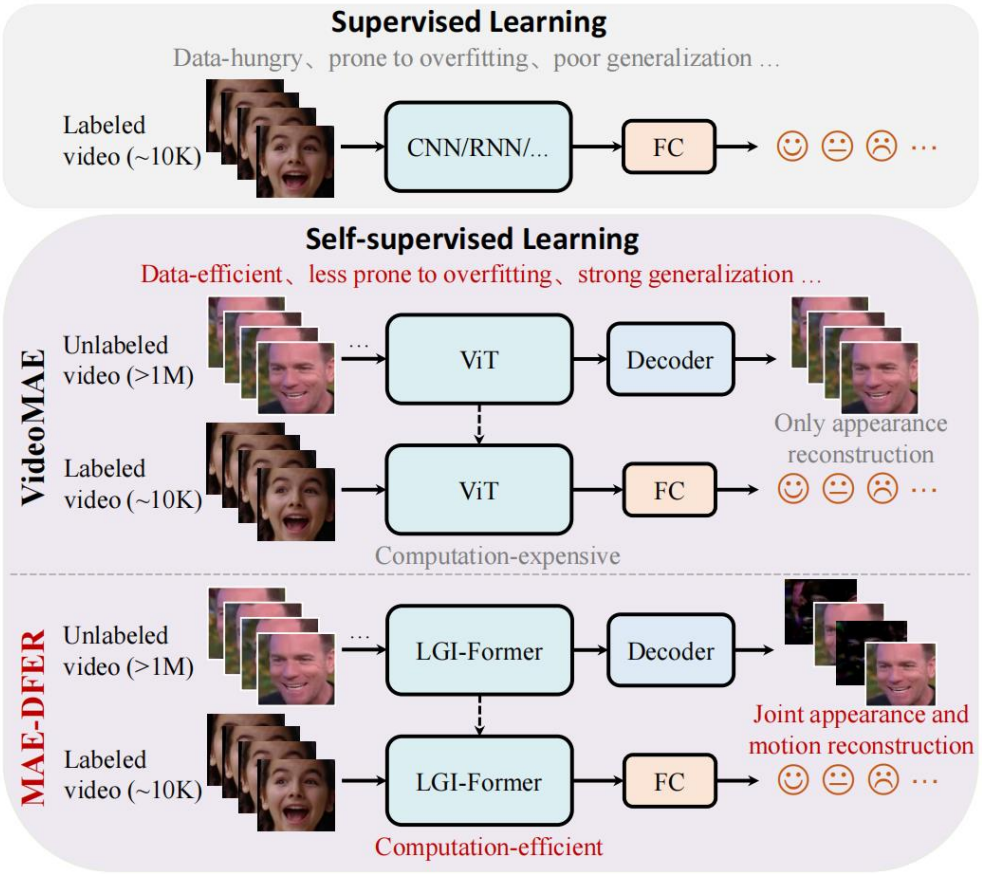

圖 3:與 VideoMAE 對比[3]

在實驗途中，我們發現到模型會因為情緒強度太小，而誤判其結果，因此我們在之後也對其做出相對應的改良。

- 宏表情辨識模型

其他基於影片的模型為 MMEmotionRecognition 其架構如下圖:

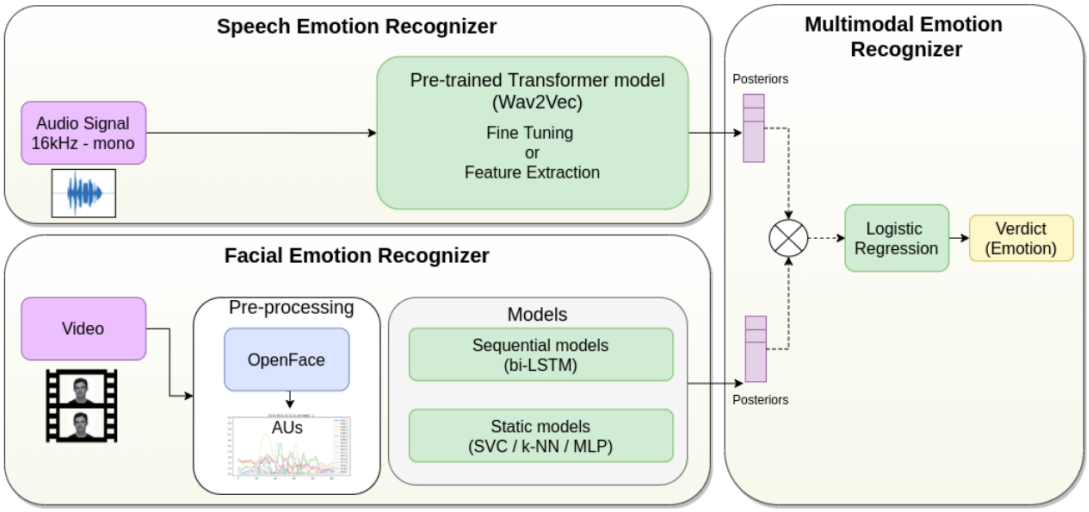

圖 4:MMEmotionRecognition 架構圖[6]

一開始我們針對影片進行 AUs(Action Units)的提取，把面部動作記錄下來後，我們選用了 static model，將每一幀記錄下來的 AUs 取平均來進行情緒的分析，發現準確度和預測速度不符合我們的需求。

針對此模型，他原本有額外的聲音情緒辨識，可能較適用於常常面對螢幕說話的實況主身上，因此相較於其他的模型，我們認為它更能夠在實況端有良好的表現，而不是用在單純觀賞體驗的觀眾身上。

- 靜態情緒辨識模型

基於圖片的模型為 POSTER-V2，其架構如下圖:

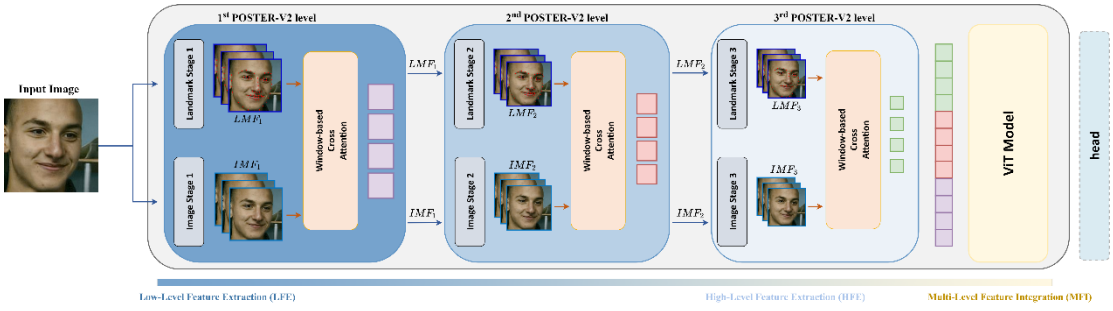

圖 5:POSTER-V2 架構圖[7]

針對圖片進行面部特徵檢測和圖像骨幹的處理，用了雙流和交叉融合的方法，進行了多尺度特徵的提取。

我們發現以多個圖像組合成影片的方式能達到一定的準確度，在 FERV39K 的準確率為 49.79%，雖略低 MAE-DFER 模型，但在各項情緒辨識上，有些略高於 MAE-DFER 模型，因此我們認為此模型能夠用來輔助與檢測其他模型的準確度。

- 微表情辨識模型

為了改善上述所提到，模型會因為情緒強度太小而誤判其結果的問題。在我們觀察到微表情辨識模型能夠較有效地偵測出面部微小變化後，我們決定將微表情辨識模型和動態情緒辨識模型進行結合。

在本次研究中我們所選用的模型為 MMNET，此模型分為兩個部分，其中以 CA_BLOCK 為主體的主要模型 CA_Module，引入 CBAM[10]來學習 Apex frame 與 Onset frame 的肌肉變化特徵，而以 ViT 為主的輔助模型 PC_Module 則用 Onset frame 來學習發生變化前，臉部相對位置的特徵，在之後透過結合這兩特徵的方式，在表情偵測中取得更好的結果，其架構如下圖:

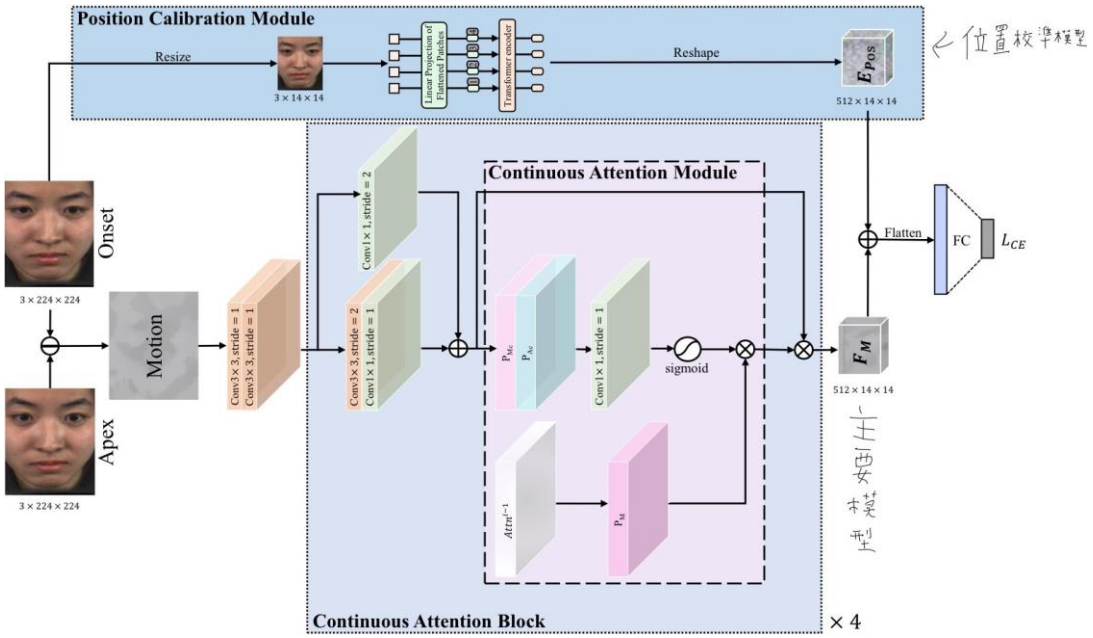

圖 6: MMNET 架構圖[8]

- 模型結合

在模型結合上，為了使 MMNET 能與 MAE-DFER 進行結合，我們將 MMNET 的 CA_Module 獨立出來，希望能藉由 CA_Module 萃取肌肉變化特徵的能力，改善 MAE-DFER 辨認微小表情能力較差的問題，並在原本的 Module 後加上全連階層使其能與 LGI-Former 萃取的特徵進行結合。接著，在特徵結合的方式上，我們嘗試了兩種方式進行結合，分別是 concatenate 和 add，最後再將結果丟入 Fully connected layer 後取得最終結果，詳細架構圖如下:

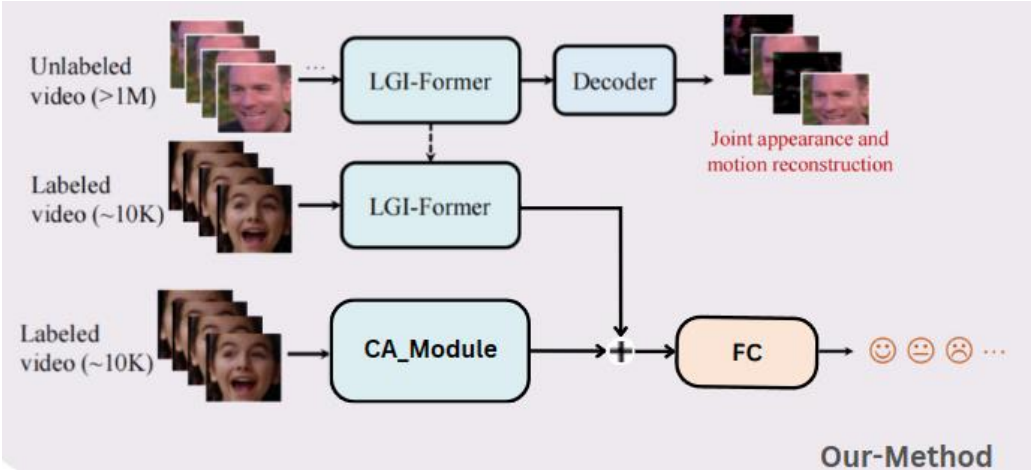

圖 7:Our Method

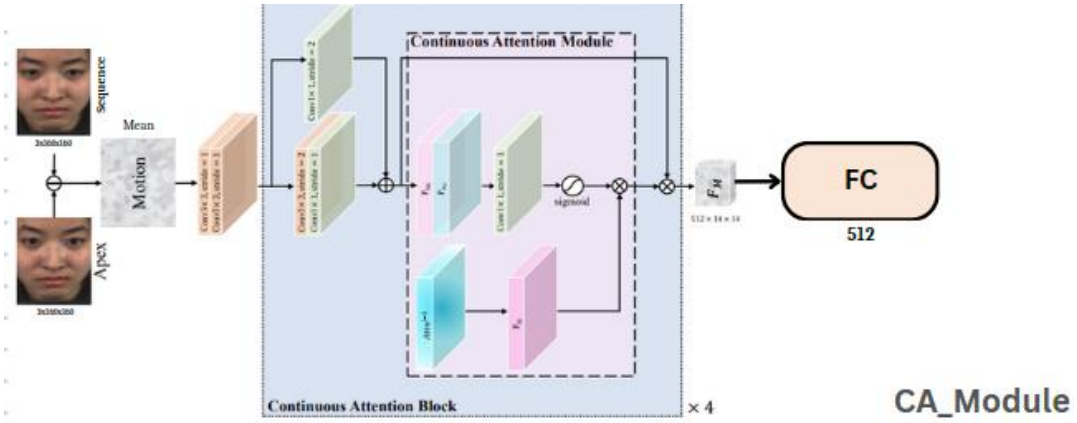

圖 8:CA_Module 架構

3.3 彙整觀眾資訊

為了達到即時反應出各個觀眾情緒變化的效果，我們利用了 Google cloud platform 的 Google Sheets API 搭配 python 語法，將 client 端得到的表情傳至雲端，接著從雲端接收所有觀眾的表情，達到 client 端之間以及對 server 端表情變化的傳輸。

3.4 使用者介面

在使用者介面的部分，我們使用 tkinter 來進行開發，原因是其跨平台性和簡易性。

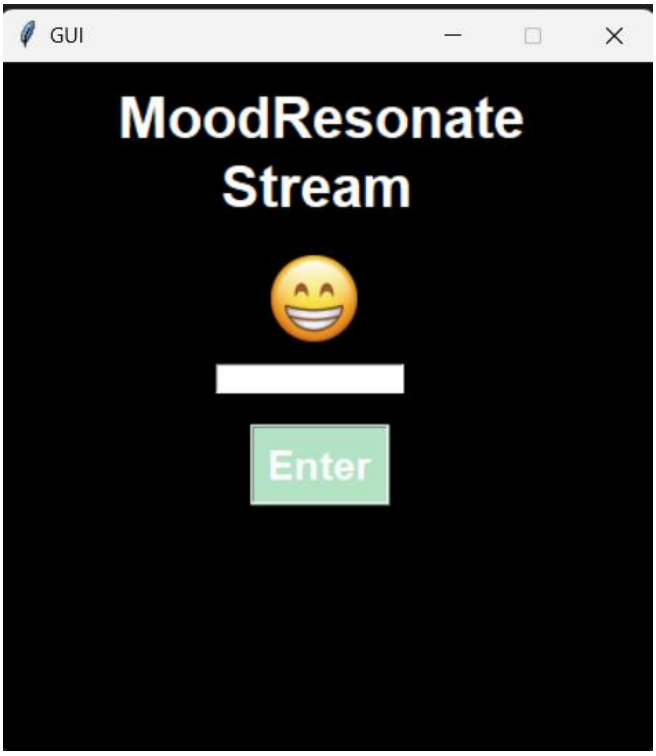

圖 9: 輸入使用者名稱

圖 10: 選擇直播頻道

圖 11: 情緒辨識功能(資料來源為:DFEW[5])

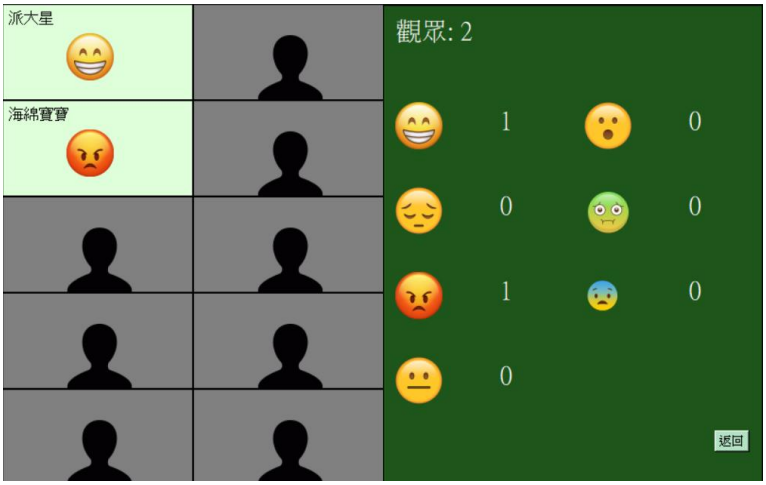

圖 12: 同步顯示所有觀眾情緒

上圖為實際使用的樣子，在左邊會看到每個使用者的情緒反應，而右方會呈現當下各個情緒出現的次數，讓觀眾可以快速地了解當前所有觀眾的情緒，藉此產生共鳴增加觀賞體驗。

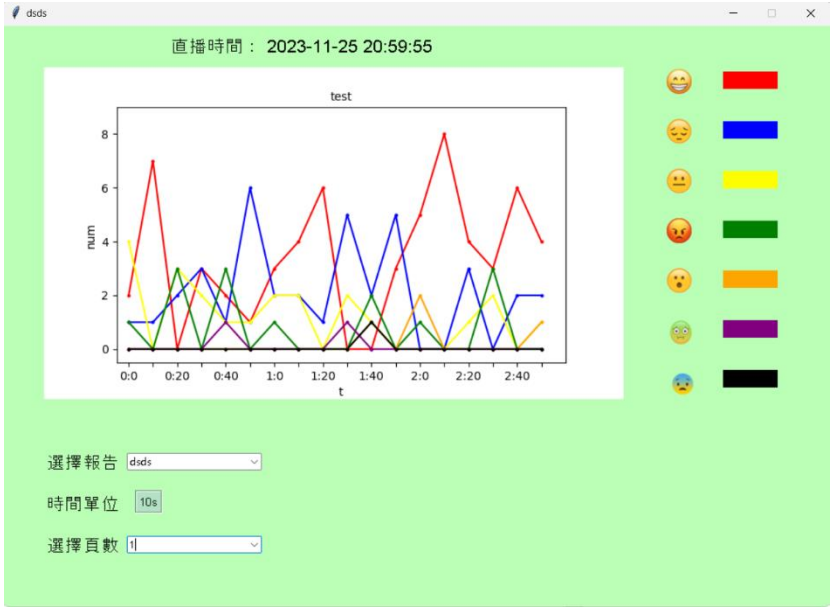

圖 13: 顯示頻道歷史紀錄

在實況直播結束後，我們利用折線圖的方式去呈現各個時間觀眾情緒的變化，目的是希望可以讓實況主在結束直播後，可以透過這個圖去了解觀眾對於實況內容的情緒反應，藉此改善自己的實況內容。

四、實驗結果

在實驗上，我們選擇 FERV39K[4]所提供的訓練和測試資料集進行驗證，該資料集具有 38935 部影片，共七個情緒分別是：Happiness、Anger、Neutral、Sadness、Surprise、Disgust、Fear。

首先，為了解決模型時常會因為情緒強度不大的關係，而誤判成其他情緒，在這邊我們先嘗試了 IAL[2]所提出的損失函數，其公式如下圖:

$$\mathcal{L}_{IA} = -\log(P_{IA}), \tag{9}$$

$$P_{IA} = \frac{e^{x_t}}{e^{x_t} + e^{x_{max}}}. \tag{10}$$

圖 14: Intensity-aware Loss[2]

其中x_t代表最後 FC 輸出之目標(Target)大小，而x_{max}代表排除目標輸出之最大值。經過我們的實驗後，結果如下表:

Loss function	**Accuracy**
Cross-Entropy	46.85%
Cross-Entropy + IAL	46.871%

表 1: Former-DFER[1]在 Ferv39k 上模型表現結果

從結果上我們發現該損失函數無法有效地解決問題。因此，我們又單獨嘗試了 MMNET[8]。為了比較其與動態面部情緒辨識方法的差異，我們同樣也使用 FERV39K 來進行訓練和測試，結果如下表:

UAR(Unweighted Average Recall)	**WAR(Weighted Average Recall)**
25%	35%

表 2: 微表情辨識模型在 Ferv39k 上結果

可以看到結果非常不好，我們認為表現不佳的原因是因為該模型設計專注在抓取臉部微小變化，跟宏表情辨識的模型不同。

Dataset	Method	UAR	WAR
FERV39K	add	**42.30%**	**52.40%**
	concatenate	42.20%	52.30%

表 3: Feature fusion 方法比較

上表是我們使用到的兩種用來將微表情模型和動態面部情緒辨識模型所萃取出的特徵進行結合所得到的結果，在實驗結果上我們發現直接相加會比串接表現稍佳。

另外，我們也測試了微表情模型各個 Module 在 add 方法下對模型表現的影響，實驗結果如下表:

CA_Module	PC_Module	#Params(M)	FLOPs(G)	UAR	WAR
×	×	85	50	42.09	51.82
✓	×	114	52	**42.30**	**52.40**
✓	✓	119	53	42.02	52.08

表 4: 各個 Module 對模型表現影響

其中 CA_Module 代表 Continuous Attention Module，而 PC_Module 代表 Position Calibration Module。

從結果可以看到在加上了微表情辨識模型中的 CA_Module，雖然造成運算量些微提升，但這個 Module 能夠讓模型在遇到情緒強度低的資料時，準確度稍微提升，且在 WAR 上有著和 SOTA 接近的表現。在之後加上 PC_Module 的測試中，可以看到單純結合 CA_Module 的表現較好，我們認為的原因是 LGI-Former 內部已經具備類似 PC_Module 的架構，而添加 PC_Module 後，導致面部位置資訊混亂，進而影響準確度。

Method		Accuracy of Each Emotion(%)						
CA	**PC**	**Happy**	**Sad**	**Neutral**	**Anger**	**Surprise**	**Disgust**	**Fear**
×	×	72.51	53.41	61.18	50.44	26.18	**15.63**	**15.31**
✓	×	75.42	55.06	60.37	**50.84**	27.43	13.49	13.46
✓	✓	75.22	54.56	**61.44**	48.42	26.80	15.20	12.53
POSTER-V2		**78.07**	**56.71**	49.34	47.00	**28.84**	13.49	12.76

表 5: 各個 Module 對各個情緒準確度的影響

另外，從上表中看出 POSTER_V2[7]在某些情緒預測上優於其他模型，但整體的結果是輸給了 MAE-DFER+MMNET(Ours)模型。

Predict(emo) = Ours(emo) * w + POSTER-V2(emo) * (1 - w)

圖 15: 模型結合公式

為了改進各情緒的準確度，我們也嘗試將 MAE-DFER+MMNET(Ours)與 POSTER-V2[7]利用 0.8 : 0.2 與 0.6 : 0.4 的權重比(w)結合起來。並使用 FERV39K 進行測試，實驗結果如下表:

Ours	**Poster**	**UAR**	**WAR**
✓	×	**42.30%**	**52.40%**
×	✓	40.89%	49.79%
0.8	0.2	40.82%	50.50%
0.6	0.4	40.83%	50.32%

表 6: 各種權重比對模型表現影響

可以發現單獨使用 MAE-DFER+MMNET(Ours)依舊比加上 POSTER-V2 後有更好的表現，即使更改權重，也沒辦法造成實質上的提升。

Method	**#Params(M)**	**FLOPs(G)**	**UAR**	**WAR**
Supervised methods				
C3D[11]	78	39	22.68	31.69
3D ResNet-18[12]	33	8	26.67	37.57
Former-DFER[1]	18	9	37.20	46.85
IAL[2]	19	10	35.82	48.54
POSTER-V2[7]	58	26	40.89	49.79
Self-supervised methods				
VideoMAE[9]	86	81	**43.33**	52.39
MAE-DFER[3]	85	50	42.09	51.82
MAE-DFER+MMNET(Ours)	114	52	42.30	**52.40**

表 7: 不同方法在 FERV39K 的結果

而表 7 結果可以看到 Self-supervised 模型表現明顯優於 Supervised 模型，而在我們自己的裝置上跑時，我們的模型相較於原本的 MAE-DFER[3]，雖然在 FLOPs 提升了 2G，但在 UAR 與 WAR 上我們的模型分別提升了 0.21%與 0.58%，更為接近 SOTA 模型。

五、結論與未來展望

在目前的結果中，我們能在有一定的準確度下即時地將觀眾的臉部情緒預測出來，將結果顯示在介面上。讓觀眾可以在觀賞直播的同時，也能知道其他觀眾的反應，產生共鳴進而提升觀賞體驗。另外，目前我們也完成了歷史紀錄的功能，透過折線圖呈現各個時間觀眾的情緒反應，讓實況主在結束直播後，能夠透過這個功能去了解觀眾對於自己實況內容的反應，藉此去改善實況內容。

在未來我們會希望能夠持續優化模型，來讓模型在遇到情緒強度低的表情時能夠表現的更好。並優化使用者與直播者介面，讓觀眾和主播的使用體驗上升。

六、參考文獻

[1] Zhao, Zengqun, and Qingshan Liu. "Former-dfer: Dynamic facial expression recognition transformer." *Proceedings of the 29th ACM International Conference on Multimedia*. 2021.

[2] Li, Hanting, et al. "Intensity-aware loss for dynamic facial expression recognition in the wild." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. No. 1. 2023.

[3] Sun, Licai, et al. "MAE-DFER: Efficient Masked Autoencoder for Self-supervised Dynamic Facial Expression Recognition." *arXiv preprint arXiv:2307.02227* (2023).

[4] Wang, Yan, et al. "Ferv39k: A large-scale multi-scene dataset for facial expression recognition in videos." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022.

[5] Jiang, Xingxun, et al. "Dfew: A large-scale database for recognizing dynamic facial expressions in the wild." *Proceedings of the 28th ACM international conference on multimedia*. 2020.

[6] Luna-Jiménez, et al. "A Proposal for Multimodal Emotion Recognition Using Aural Transformers and Action Units on RAVDESS Dataset." Applied Sciences, 12(1), 327.

[7] Mao, Jiawei, et al. "POSTER V2: A simpler and stronger facial expression recognition network." *arXiv preprint arXiv: 2301.12149* (2023).

[8] Li, Hanting, et al. "MMNet: Muscle motion-guided network for micro-expression recognition." *arXiv preprint arXiv:2201.05297* (2022).

[9] Tong, Zhan, et al. "Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training." Advances in neural information processing systems 35 (2022): 10078-10093.

[10] Woo, S., Park, J., Lee, J.-Y., & Kweon, I. S., "CBAM: Convolutional Block Attention Module," arXiv preprint arXiv:1807.06521, 2018.

[11] Tran, Du, et al. "Learning spatiotemporal features with 3d convolutional networks." Proceedings of the IEEE international conference on computer vision. 2015.

[12] Hara, Kensho, Hirokatsu Kataoka, and Yutaka Satoh. "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?." Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. 2018.