基於螢幕的視線追踪模型

Screen-Based Gaze Tracking Model

潘品齊
電機工程學系
國立中正大學
嘉義，臺灣
pinchi1609@gmail.com

陳俊騰
電機工程學系
國立中正大學
嘉義，臺灣
anc891203@gmail.com

Abstract—人眼視線追蹤一直是電腦視覺領域最具挑戰性的問題之一，該研究已被廣泛的使用在各個領域中，例如駕駛人輔助、虛擬實境和人機互動等，而其中駕駛人輔助應用是能夠藉由視線追蹤模型預測車輛駕駛人視線範圍並評估其狀態，在駕駛安全中起著重要作用。然而，傳統視線追蹤模型的視線預測方法主要是由眼睛狀態來估計其視線範圍，該方法在穿戴式視線追蹤設備上能夠良好運作，不過其缺點是穿戴式設備成本過高，若改為使用非穿戴式設備進行視線追蹤，便能夠有效降低成本，但也會因為頭部姿勢的運動而影響視線預測效果。本研究提出了一種基於外觀的頭部姿勢估計並結合視線追蹤的預測方法，除此之外我們還使用機器學習中的決策樹演算法來預測駕駛人正在觀看的視線區域。

***Keywords*—視線追蹤、頭部姿勢估計、人眼追蹤、視線分析、決策樹**

I. 緒論

根據統計交通事故的主要原因多數是來自於分心駕駛，其中包括駕駛人使用手機、手中拿東西、抽菸或進食、調整音響或空調等，可見駕駛人的注意力對於安全駕駛極為重要。而駕駛人當前的注意力與其視線方向有密切關係，因此研究駕駛人的視線方向已經廣泛應用於駕駛人的狀態和注意力檢測當中。視線追蹤裝置通常分為穿戴式以及非穿戴式設備，穿戴式視線追蹤裝置雖然有較高的準確率，但由於其生產成本較高，並且在現實生活中不利於使用，故不常被用來投入實際應用當中。相反的，非穿戴式視線追蹤系統能夠大幅降低製造成本，並且有著更高的靈活性，該設備在實際應用上的價值遠高於穿戴式設備。使用非穿戴式設備進行視線預測時，由於頭部姿勢貢獻了主要的視線方向，因此這些方法中的大多數將頭部方向視為粗略視線方向的近似值。但在真實生活的駕駛中，許多駕駛人在看目標時會同時移動頭部和眼睛。 Tawari 等人 [1] 比較了實驗中僅使用頭部姿勢以及同時使用頭部和眼睛姿勢的視線預測性能。若能有效結合頭部與眼睛姿勢預測，其準確率也會顯著提高。 Fridman 等人 [2] 進一步指出，駕駛人頭部保持靜止狀態時加入視線預測的準確率比頭部移動幅度較大時加入視線預測時提高的更多。

本研究將會探討如何從單張圖像中預測駕駛人頭部姿勢及眼睛姿勢，並有效結合多個特徵來預測其視線區域。為了預測頭部姿勢，我們利用了一種基於 landmark-free 的頭部姿勢預測方法，該方法使用帶有九個參數的旋轉矩陣來表示回歸準確的頭部方向。該旋轉矩陣可以實現全姿勢回歸，而不會出現 Gimbal Lock 的問題。在預測眼睛姿勢的部分，我們利用一種使用多重損失的方法從圖像預測 3D 視線角度。並使用兩個全連接層獨立回歸每個視線角度(Yaw, Pitch)，以提高每個角度的預測準確率來預測 3D 的視線方向。最後，我們藉由預測出的頭部與眼睛姿勢，以及臉部座標、與鏡頭距離等多個特徵，利用機器學習中的決策樹演算法模型，來預測駕駛人正在觀看的視線區域。

接下來第二章將介紹臉部及視線預測方面的相關研究，第三章會描述我們的實驗架構模型與改善方法，在第四章將介紹實驗的訓練流程、實驗結果及結果探討，並在最後第五章進行總結及本研究的未來展望。

II. 背景知識與文獻探討

視線追蹤已經被廣泛應用於駕駛人的狀態和注意力檢測當中，其關鍵技術對於安全駕駛有獨特的應用價值，視線追蹤對於人車互動具有直接性及雙向性的特點，能提供更多互動的可能性。本研究重點在於如何有效預測頭部及視線方向，所以本章節將依序介紹頭部姿勢預測及視線方向預測在先前研究中的成果，詳細敘述如下：

A. 旋轉表示(Rotation Representation)

在處理角度預測時的關鍵方法是使用適當的旋轉表示。歐拉角(Euler angles)是最常使用也是最方便的旋轉表示方式，如圖 1 所示。然而，這種表示方式並不是最好的，因為它會受到 Gimbal Lock 的影響，在這種情況下，對於相同的頭部姿勢外觀會有多個旋轉參數。另一種旋轉表示的方法是四元數(Quaternion) ，如圖 2 所示。雖然四元數不受 Gimbal Lock 的影響，但由於其對映對稱性，在學習全方位的頭部姿勢時，可能會導致預測性能下降。在預測全方位的頭部姿勢時，最好的旋轉表示方式是使用旋轉矩陣，如圖 3 所示，它是一個連續的表示方式，並且每次旋轉都有不同的參數。

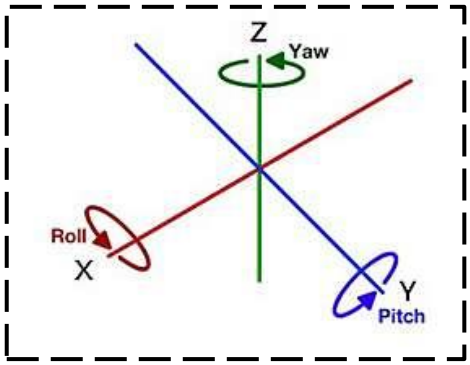

圖1、歐拉角(Euler angles)表示方式

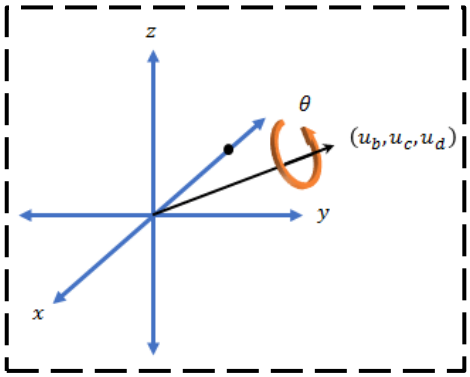

圖2、四元數(Quaternion)表示方式

$$\mathcal{R}_x(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & -\sin\theta_x \\ 0 & \sin\theta_x & \cos\theta_x \end{bmatrix}$$

$$\mathcal{R}_y(\theta_y) = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y \\ 0 & 1 & 0 \\ -\sin\theta_y & 0 & \cos\theta_y \end{bmatrix}$$

$$\mathcal{R}_z(\theta_z) = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 \\ \sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

圖3、旋轉矩陣表示方式

B. 頭部姿勢預測(Head Pose Predictions)

頭部姿勢預測目前常用的方法通常分為 landmark-based 與 landmark-free。landmark-based [3] 的方法在最開始會先檢測臉部關鍵點，並在後續步驟中通過建立這些關鍵點和 3D 頭部模型之間的對應關係來恢復 3D 頭部姿勢。雖然這種方法可以產生非常準確的結果，但它高度依賴於對關鍵點位置的正確預測。因此，由遮擋和極端旋轉引起的劣質關鍵點會損害準確的頭部姿勢估計。而 landmark-free 的方法如 HopeNet [4] ，通過直接估計頭部姿勢克服了這個問題，這種方法通常有助於深度神經網路將方向預測制定為基於外觀的任務。

基於外觀的頭部姿勢預測， 6DRepNet [5] 提出了一種 landmark-free 的端到端頭部姿勢預測方法，該方法通過引入旋轉矩陣形式來解決模糊旋轉標籤的問題，九參數矩陣可以實現全姿勢預測。除此之外該論文還提出一個連續的 6D 矩陣表示，該表示在後續任務中能夠轉換為旋轉矩陣，以實現高效和穩定的預測方式，如圖 4 所示。

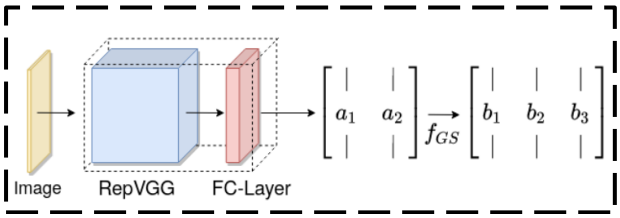

圖4、6DRepNet 頭部姿勢預測方法概述

藉由 6DRepNet 所提出的方法可以學習完整的旋轉外觀，這與以前將姿勢預測限制在窄角度以獲得令人滿意的結果的方法相反，除此之外該方法所使用的損失函數是使用他們提出的 Geodesic Distance-Based Loss 而不是常用的均方誤差損失函數，如圖 5 所示。

$$L_g = cos^{-1}\left(\frac{tr(R_p R_{gt}^T) - 1}{2}\right)$$

圖5、Geodesic Distance-Based Loss

雖然頭部姿勢相關任務常用的損失函數是 L2-Norm，但若使用 Frobenius Norm 測量兩個矩陣之間的距離將會破壞 SO(3) 流形幾何。相反，兩個 3D 旋轉矩陣之間的最短路徑在幾何上被解釋為 Geodesic Distance，該研究也已證明利用 Geodesic Distance-Based Loss 來訓練網路能夠取得更好的結果。

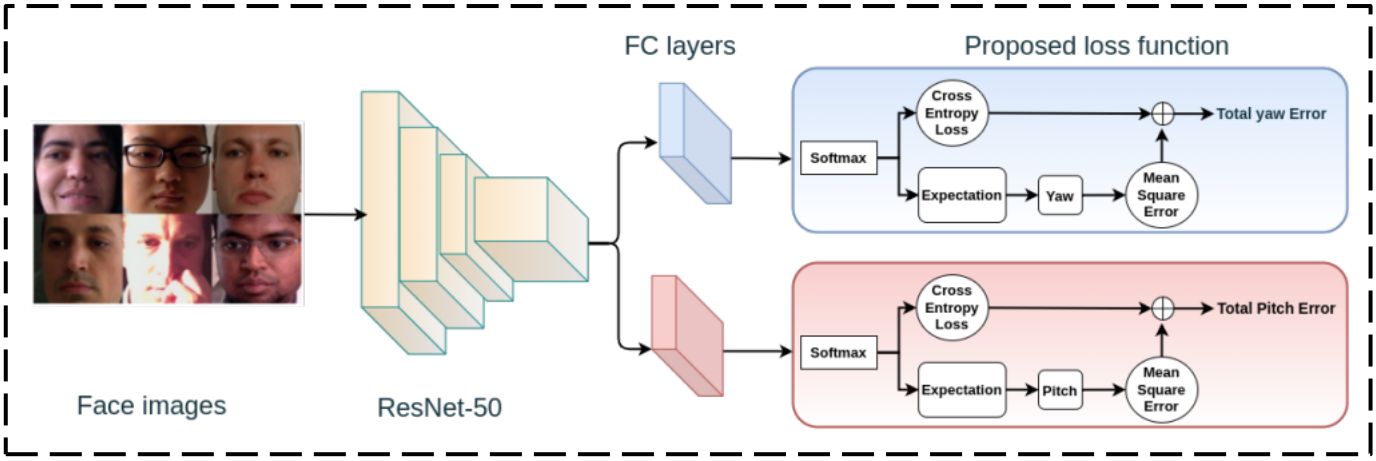

圖6、L2CS-Net 視線追蹤預測方法概述

C. 視線追蹤(Gaze Tracking)

人眼視線方向一直是人機互動與虛擬實境等各種應用中使用的關鍵線索。雖然深度學習藉由卷積神經網路方法在預測視線方向方面取得了顯著的進展，然而由於眼睛外觀、光線條件的獨特性以及頭部姿勢和視線方向的多樣性，在非穿戴式設備上預測視線仍是一個具有挑戰性的問題。

大多數基於 CNN 的注視估計模型將 3D 視線預測為球坐標中的視線方向(Yaw、Pitch)。訓練損失函數通常採用均方誤差(MSE Loss)來懲罰網路。L2CS-Net [6] 提出了一種基於 CNN 的視線追蹤模型，用於在不受約束的環境中預測視線方向。該網路使用 ResNet50 作為主要的網路架構，在訓練方面分別回歸 Yaw 與 Pitch 角度，以提高每個角度預測的準確性，這將提高整體視線預測性能。除此之外，該網路使用兩個相同的損失，每個損失函數是一個 Cross Entropy Loss 和 Mean-Squared Error 的組合，以改進網路學習並增加其泛化性。在視線預測方面，L2CS-Net 並沒有直接預測連續的視線角度，而是使用具有 Cross Entropy 的 Softmax Layer 來預測離散的視線分類，並使用離散的視線預測結果轉化為連續的視線角度，最後在輸出中加入均方誤差以改進視線預測，整體的視線追縱方法如圖 6 所示。

D. 決策樹(Decision Tree)

決策樹會根據訓練資料產生一棵規則樹，依據訓練出來的規則來對新樣本進行預測。決策樹演算法可以使用不同的方式來評估分枝的好壞，例如 Information Gain、Gain Ratio、Gini Index 等。依據訓練資料找出合適的規則，最終生成一棵規則樹來決策所有事情，其目的使每一個決策能夠使訊息增益最大化，如圖 7 所示。決策數在越上層的決策中會以對最終決策影響較大的特徵先進行第一次的決策判斷。接著越往下層會再從這些特徵中尋找最適合的決策因子，直到設定的最大深度即停止樹的生長。

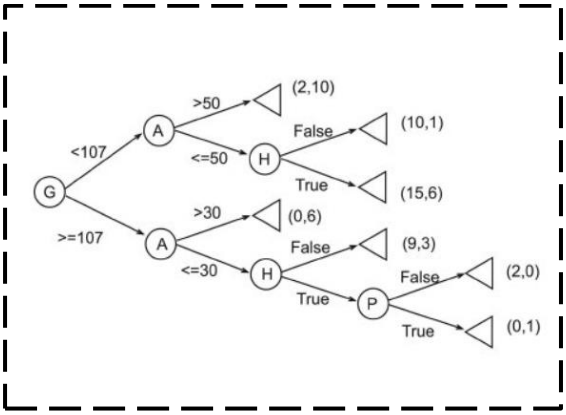

圖7、決策樹示意圖

決策樹的生成是以貪婪法則來決定每一層的問題，目標是讓分類過後的每條分枝都能夠更明顯的表示所屬類別。然而要如何去判斷每次決策的好壞，就必須依靠亂度的評估指標。客觀的標準來決定決策樹的每個分支非常重要，因此我們需要一個評斷的指標來協助決策。決策樹演算法可以使用不同的指標來評估分枝的好壞，常見的決策亂度評估指標有 Information Gain、Gain Ratio、Gini Index 等。該評估方法的目標是從訓練資料中找出一套決策規則，讓每一個決策能夠使訊息增益最大化。

決策樹的訓練過程就是不斷的尋找特徵進行決策，透過這些決策盡量的使這些資料被分為同一個類別，且試著讓混亂程度越小越好。決策樹的深度越深雖然能增加準確率，但也可能因此造成過擬合的問題。一棵訓練好的決策樹模型能夠視覺化其結構，相對的可解釋性較高。此外與其它的機器學習模型相比，由於其樹狀結構在進行機器學習時每個決策階段都相當明確，執行速度也就相當迅速，適合應用在即時應用中。

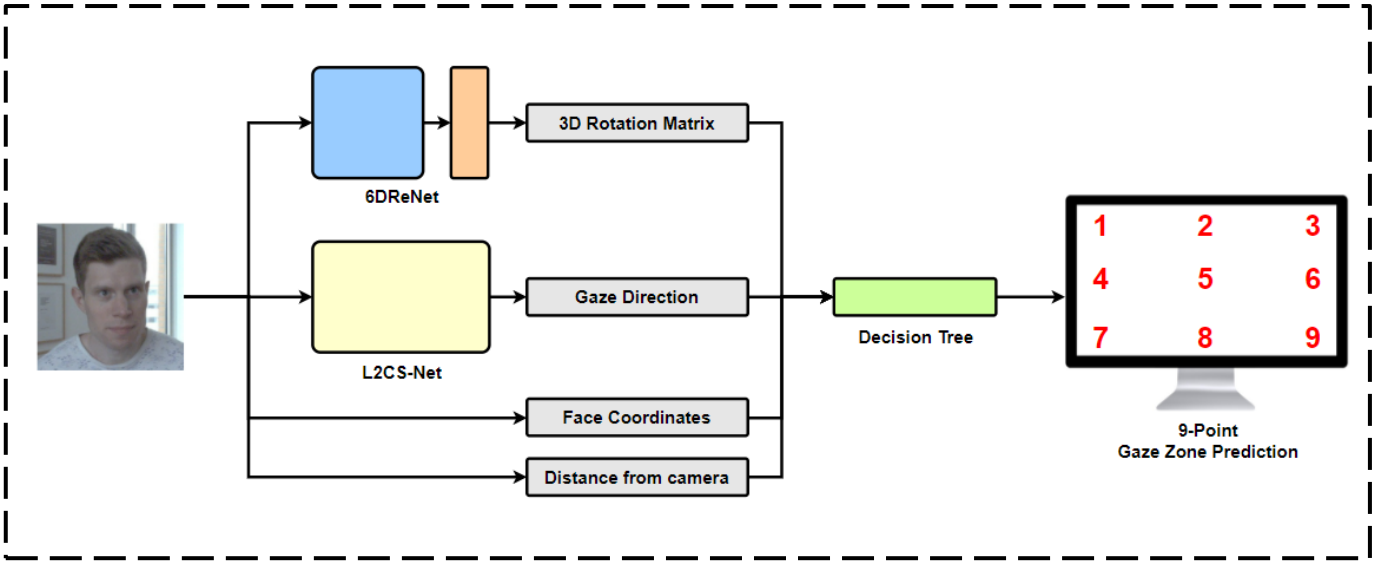

圖8、基於螢幕的視線追踪模型（Screen-Based Gaze Tracking Model）方法概述

III. 研究方法

在第二章所介紹的頭部姿勢預測模型中，我們以 6DRepNet 作為訓練模型進行改善，使用不同切入點試圖強化模型效能，如：修改模型架構、使用不同損失函數進行訓練等。在人眼視線預測方面，我們使用 L2CS-Net 作為訓練網路，同樣使用不同的深度學習網路替換主要網路架構進行優化。本研究基於 6DRepNet 與 L2CS-Net 分別做為臉部姿勢預測及人眼視預測模型基礎，我們提出基於螢幕的視線追踪模型(Screen-Based Gaze Tracking Model)，該模型主要利用 6DRepNet 與 L2CS-Net 的輸出作為特徵，並加入使用者的臉部座標以及人臉與非穿戴式鏡頭的距離作為複數特徵，利用決策樹演算法學習預測使用者正在觀看的螢幕區域以代替預測駕駛人正在觀看的視線區域，整體架構如圖 8 所示。

A. 資料集(Datasets)

在頭部姿勢預測的訓練中，我們主要使用的訓練資料集為 300W-LP [7] ，該資料集由 66225 個臉部樣本組成，通過圖像翻轉進一步增強為 122415 張樣本。在測試方面我們使用 AFLW2000 資料集 [8] ，該資料集包含來自 AFLW 資料集的前 2000 張圖像，這些圖像用真實 3D 人臉和對應的 68 個地標進行了標註，它具有較大變化、不同照明和遮擋條件的樣本。在本研究中我們使用 300W-LP 資料集進行模型訓練，並在 AFLW2000 中的 1969 張圖像上進行評估。

在人眼視線預測的訓練中，我們使用兩個不受環境限制的資料集來訓練和評估我們的模型：Gaze360 和 MPIIGaze。Gaze360 [9] 提供最廣泛的 3D 視線標註，最大範圍為 360 度。它包含 238 個不同年齡、性別和種族的受試者，並且在圖像收集方面是使用多攝像頭系統在不同的室內和室外環境拍攝。MPIIGaze [10] 提供了 213659 張來自 15 名受試者在幾個月的日常生活中拍攝的圖像。因此，它包含具有不同背景、時間和照明的圖像，使其適用於不受限制的視線預測。在圖像收集方面，它使用軟體收集圖像，該軟體要求參與者查看筆記本電腦上隨機移動的點。

在預測使用者正在觀看的螢幕區域訓練中，我們將電腦螢幕依照九宮格劃分為九個區域，並透過筆記型電腦螢幕鏡頭蒐集我們自製的資料集，每個區域蒐集 200 張圖像進行訓練，共有 1800 張訓練圖像，並在每個區域使用 50 張圖像進行測試，共有 450 張測試圖像。

B. 資料預處理(Data Preprocessing)

對於 300W-LP 與 AFLW2000 資料集，我們遵循其他方法 [11, 12] 的預處理策略，只保留歐拉角在 -99° 和 99° 之間的圖像。

對於 Gaze360 與 MPIIGaze 資料集，我們依照 [13] 中的相同方法對兩個數據集中的圖像進行正規化。這個過程對虛擬相機應用旋轉和平移，以消除頭部的滾動角度，並保持虛擬相機和臉部中心之間的距離相同。此外，我們將每個數據集中的連續視線方向(Yaw, Pitch)分成帶有二進制標籤的離散表示，以便根據視線標註的範圍進行分類。因此，兩個數據集都有兩個不同的目標標註：連續標籤和離散標籤使它們適合用於組合回歸和分類損失。

對於我們自製的螢幕視線區域資料集，我們簡單將其分為九個資料夾，對應到電腦螢幕中的

九個區域。使用訓練好的臉部及視線預測模型估計臉部角度與視線方向，並加入臉部座標位置以及與鏡頭的距離共八個特徵。每張圖片的八個特徵與其對應到的資料夾名稱便是決策樹模型的訓練輸入與輸出資料。

C. 6DRepNet 模型架構(6DRepNet Architecture)

我們所使用的模型架構如圖 4 所示。我們利用 Vision Transformer 與 Swin Transformer 網路架構，替換原有的 RepVGG 架構進行訓練。在模型的輸出方面，我們依照 Zhou 等人的方法 [14] 並通過簡單地刪除旋轉矩陣的最後一列向量，在旋轉表示內部執行 Gram-Schmidt Mapping。這將 3 x 3 旋轉矩陣減少為 6D 旋轉表示，如圖 9 所示。根據研究顯示這種作法為直接回歸引入了更小的誤差。

$$g_{GS} = \left(\begin{bmatrix} | & | & | \\ a_1 & a_2 & a_3 \\ | & | & | \end{bmatrix} \right) = \begin{bmatrix} | & | \\ a_1 & a_2 \\ | & | \end{bmatrix}$$

圖9、將 g_{GS} 映射到僅刪除最後一列的表示空間

雖然我們的網路模型只預測 6 個參數，但是我們可以藉由圖 10 的公式，將其映射回 3 x 3 的旋轉矩陣當中，剩餘的列向量由 Cross Product 決定，進而確保得到的旋轉矩陣滿足正交性。

$$f_{GS} = \left(\begin{bmatrix} | & | \\ a_1 & a_2 \\ | & | \end{bmatrix} \right) = \begin{bmatrix} | & | & | \\ b_1 & b_2 & b_3 \\ | & | & | \end{bmatrix}$$

$$b_1 = \frac{a_1}{||a_1||}$$

$$b_2 = \frac{u_2}{||u_2||}, u_2 = a_2 - (b_1 \cdot a_2) b_1$$

$$b_3 = b_1 \times b_2$$

圖10、藉由 6 參數映射回旋轉矩陣

在後面的研究中，我們將使用如圖 5 中的 Geodesic Distance-Based Loss 作為我們的神經網路損失函數來計算預測和真實方向之間的準確距離信息，並與 L1 Loss、MSE Loss 進行比較。

D. L2CS-Net 模型架構(L2CS-Net Architecture)

L2CS-Net 是基於分類和回歸損失提出的網路架構，它將人臉圖像輸入至 ResNet50，以從圖像中提取空間視線特徵。與先前作法不同的是 L2CS-Net 將提取後的空間視線特徵輸入進兩個全連接層分別預測每個角度，這兩個全連接層共享相同的卷積層。此外，我們使用兩個損失函數，用來獨立回歸每個視線角度，因為它有兩個通過網路反向傳播的訊號，使用這種方法便能改善網路學習。

對於全連接層的每個輸出，我們首先使用 Softmax 層將網路輸出 Logits 轉換為概率分佈。利用交叉熵損失來計算輸出概率和目標離散標籤之間的離散分類損失。接著我們計算概率分佈的期望以獲得細粒度的視線預測。最後我們計算該預測的均方誤差並將其添加到分類損失中，詳細架構如圖 6 所示。

E. 決策樹模型架構(Decision Architecture)

為了預測使用者正在觀看的螢幕區域，我們將臉部姿勢預測出的 Yaw、Pitch、Roll 值，以及人眼視線預測出的 Yaw、Pitch 值，加入臉部 x 軸座標與 y 軸座標以及與鏡頭的距離作為輸入訓練我們的決策樹模型。我們透過人臉偵測模型偵測臉部五個點，包含左眼、右眼、鼻子、嘴唇左側與右側，並利用鼻子的座標表示臉部 x 軸與 y 軸座標。而人臉與鏡頭距離的計算方式我們是採用圖 11 的計算方式。

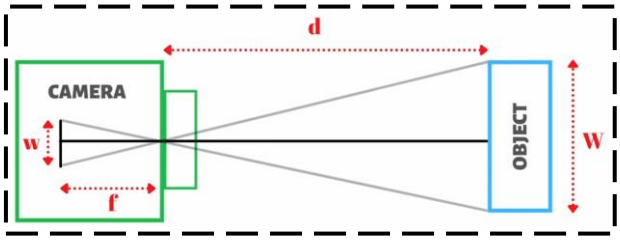

圖11、物體與鏡頭距離示意圖

其中 f 為焦距、w 為物體在螢幕上的大小、d 為物體與攝影機的距離、W 為物體在真實世界中的大小。在本研究中，我們將 W 設定為兩眼之間的距離，根據統計人體兩眼的平均距離為 6.3 公分。而 w 為兩眼之間在螢幕上的距離，以像素為單位表示。為了計算物體與鏡頭的距離，必須先求得相機焦距 f 。在本研究中我們可以藉由固定物體與攝影機的距離 d，透過圖 12 中的公式求得相機焦距 f 。

Focal Length	Distance Length
f = (w * d) / W	d = (W * f) / w

圖12、物體與鏡頭距離計算方式

由於相機的焦距不會變動，所以不需要實時計算焦距大小。在求得相機焦距 f 後，我們便可以藉由兩眼之間的距離估算出人臉與攝影機的距離。值得注意的是由於我們用來估算相機焦距的兩眼距離 W 是透過統計的平均距離，所以在計算人臉與攝影機的距離時，仍與實際的距離有些誤差，不過在將其應用於決策樹的訓練上來說，訓練效果仍然非常顯著。此外，在決策樹的訓練方面，為了避免決策樹的深度太深導致過擬合的問題，我們將決策樹的最大深度設為 11，經過訓練後的決策樹架構如圖 13 所示。

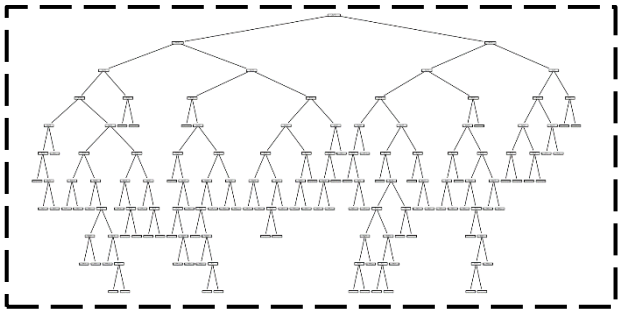

圖13、決策樹模型架構

IV. 實驗結果與討論

A. 實驗一：6DRepNet 不同網路架構之性能

在頭部姿勢預測方面，我們選擇使用 Vision Transformer 以及 Swin Transformer 網路模型與 6DRepNet 原本的主要網路 RepVGG 進行比較，我們實作的結果如圖 14 所示。

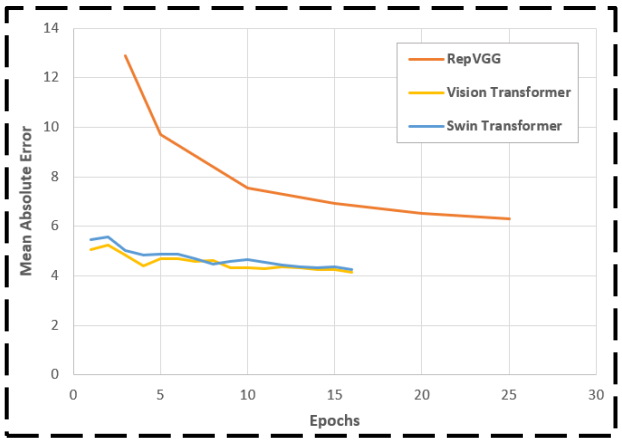

圖14、6DRepNet 使用不同架構下的模型表現

從圖 14 中可以看出，雖然在 6DRepNet 的原始論文中表明，他們的方法可以達到平均絕對誤差值為 3.97，但經過我們實際訓練後，該網路的平均絕對誤差值僅達到 6.3 便已收斂，這種性能的網路模型顯然不能夠應用於現實生活中。於是我們將其中的 RepVGG 網路架構，替換為 Vision Transformer 與 Swin Transformer ，為了更好的比較兩個模型，我們所替換的模型皆為 base 大小，並且在 ImageNet 上經過預訓練。根據我們的研究表明，基於 Transformer 的 6DRepNet 模型在經過一次的訓練後，便能將平均絕對誤差值降低至 6 以下，並且效能明顯優於基於 RepVGG 的 6DRepNet 模型。我們最後分別將基於 Vision Transformer 與 Swin Transformer 的模型訓練到平均絕對誤差值為 4.1563 與 4.2513，由於基於 Vision Transformer 的模型在我們的研究中表現最好，在後面的研究與應用中我們皆是使用基於 Vision Transformer 的 6DRepNet 網路模型。

B. 實驗二：6DRepNet 不同損失函數之性能

同樣在頭部姿勢預測方面，我們選擇使用 MSE Loss 及 L1 Loss 與 6DRepNet 原本的損失函數 Geodesic Loss 進行比較，實作結果如圖 15 所示。

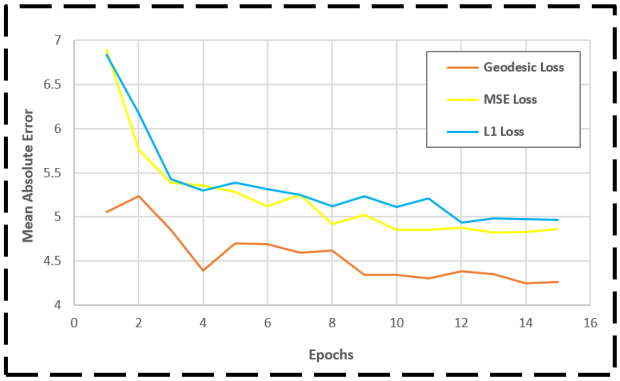

圖15、6DRepNet 使用不同損失函數的模型表現

在 6DRepNet 原始論文中有提到 Geodesic Distance-Based Loss 能更好的計算兩個矩陣之間的距離。根據我們的研究表明，MSE Loss 與 L1 Loss 在訓練 6DRepNet 模型方面的結果差異並不大，從圖 15 也可看出 Geodesic Distance-Based Loss 明顯比一般常用的損失函數 MSE Loss 與 L1 Loss 能獲得更令人滿意的結果。

C. 實驗三：L2CS-Net 不同網路架構之性能

在人眼視線預測方面，我們選擇使用 Vision Transformer 以及 Swin Transformer 網路模型替換 L2CS-Net 原本的主要網路 ResNet50 進行比較，我們實作的結果如表 1 所示。

Methods	MAE
L2CS-Net (Paper Result)	10.41
L2CS-Net (ResNet50)	10.98
L2CS-Net (Vision Transformer)	11.77
L2CS-Net (Swin Transformer)	11.65

表1、不同架構下之 L2CS-Net 模型表現

從表 1 中可以看到雖然在 L2CS-Net 的原始論文中表明，該方法可以在 Gaze360 資料集上達到平均絕對誤差值為 10.41，經過我們實際訓練後該網路的平均絕對誤差值能夠達到 10.98，這與原始論文結果相近。為了追求更好的性能，我們同樣比較了基於 Transformer 的模型架構。我們使用與頭部姿勢預測相同規格的 Vision Transformer 與 Swin Transformer 模型，根據我們的研究表明基於 Transformer 的模型，在視線預測方面仍不及基於 CNN 的網路模型且訓練所花費的時間也更長。我們最後使用 L2CS-Net 中的原始網路架構作為在即時視線追蹤的預測模型。

除了 Gaze360 外，我們也在 MPIIGaze 資料集上進行訓練，值得注意的是雖然我們在 MPIIGaze 上能夠將模型訓練至與原始論文相差不遠的結果，但我們在將其應用於即時視線追蹤時，該模型無法有效辨識即時人眼視線追蹤。我們推測該問題是由於 Gaze360 資料集在訓練時是輸入整張人臉圖片至模型來預測視線方向，而 MPIIGaze 資料集在訓練時只有輸入眼睛圖片，由於 MPIIGaze 資料集多為角度較小的眼睛圖像，故在實際預測大角度的視線方向時表現較差，我們在後續的即時視線預測上，皆是使用透過 Gaze360 資料集訓練的人眼視線預測模型。

D. 實驗四：消融研究

對於消融研究，我們將不同數量的特徵當作決策樹的訓練資料進行測試。表 2 展示了不同特徵對於模型性能的影響，從表中可以簡單的看出當特徵數量越多時，決策樹模型的性能也相對提升。

# Features	Gaze Pitch	Gaze Yaw	Face Pitch	Face Yaw	Face Roll	Face X	Face Y	Face Depth	Accuracy
5	✓	✓	-	-	-	✓	✓	✓	81.78%
6	✓	✓	✓	✓	✓	-	-	✓	81.78%
7	✓	✓	✓	✓	✓	✓	✓	-	82.00%
8	✓	✓	✓	✓	✓	✓	✓	✓	**83.11%**

表2、使用不同特徵數量訓練決策樹之結果

我們在頭部姿勢與人眼視線預測外，加入臉部座標以及人臉與鏡頭的距離作為參數訓練我們的決策樹模型。若模型訓練時不考慮臉部座標，我們的決策樹模型表現會下降約一個百分點；而當模型不考慮臉部座標或角度進行訓練時，決策樹模型的表現會下降更多，這也證明了我們最初的想法，即頭部姿勢的運動會直接影響視線追蹤模型的性能，故我們在訓練視線追蹤模型時，應該要始終加入頭部姿勢預測結果以提升模型效能。

E. 實驗五：螢幕視線區域預測結果

圖 17 展示了我們基於螢幕的視線追蹤模型預測結果，以混淆矩陣表示。從圖中可以看到我們的模型在每個區域的預測上，最可能會誤判的類別為其上方與下方的區域。根據我們的研究發現越靠近鏡頭的區域，其準確率比其他區域還高出 10%。此外，螢幕正中央的位置更有可能比其他區域更容易產生誤判。在本研究中，我們的基於螢幕之視線追蹤模型經過訓練後可達到 83.11% 的準確率。

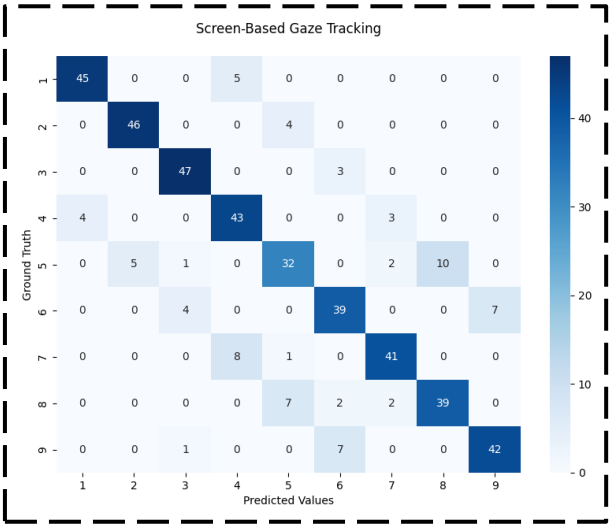

圖17、9-Point 螢幕視線區域預測結果

V. 結論與未來展望

在本研究中我們提出了一種基於螢幕的視線追踪模型(Screen-Based Gaze Tracking Model)，改善了非穿戴式設備在進行視線預測時會因為頭部姿勢而導致視線預測產生誤差的問題，並且我們也分析了在不同架構與損失函數的訓練下，頭部姿勢與人眼視線預測模型的性能表現，根據我們的研究表明在頭部姿勢預測方面基於 Transformer 的網路模型，比起基於 CNN 的網路模型表現更好。而在人眼視線預測方面，基於 CNN 的網路模型表現較優異。在螢幕視線區域預測方面，除了頭部姿勢與人眼視線預測外，透過加入人臉座標以及人臉與鏡頭的距離當作決策樹的訓練參數，能使模型在預測使用者所觀看的螢幕區域時能更準確。

在未來展望方面，本研究中所使用的基於 Transformer 的基本模型架構，若是能夠分別依據頭部姿勢與人眼視線的特徵微調模型架構，相信能夠獲得更好的結果。此外，由於 Transformer 中的注意力機制需要較為龐大的計算量，若能有效輕量化網路模型，相信在即時的預測上也能更順暢，甚至能夠在邊緣裝置上運作。

參考文獻

[1] A. Tawari, K. H. Chen and M. M. Trivedi, "Where is the driver looking: Analysis of head, eye and iris for robust gaze zone estimation," 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), 2014, pp. 988-994.

[2] L. Fridman, J. Lee, B. Reimer, T. Victor. ""Owl" and "Lizard": Patterns of Head Pose and Eye Pose in Driver Gaze Classification." arXiv preprint arXiv:1508.04028, 2015.

[3] P. Werner, F. Saxen, and A. Al-Hamadi, "Landmark based head pose estimation benchmark and method," in 2017 IEEE International Conference on Image Processing (ICIP), 2017, pp. 3909–3913.

[4] N. Ruiz, E. Chong and J. M. Rehg, "Fine-Grained Head Pose Estimation Without Keypoints," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2018, pp. 2155-215509.

[5] T. Hempel, A. A. Abdelrahman, and A. Al-Hamadi, "6D Rotation Representation For Unconstrained Head Pose Estimation," arXiv preprint arXiv:2202.12555.

[6] A. A. Abdelrahman, T. Hempel, A. Khalifa, A. Al-Hamadi, "L2CS-Net: Fine-Grained Gaze Estimation in Unconstrained Environments," arXiv preprint arXiv:2203.03339.

[7] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Li, "Face Alignment Across Large Poses: A 3D Solution," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 146–155, 2016.

[8] X. Zhu, Z. Lei, J. Yan, D. Yi, and S. Z. Li, "High-Fidelity Pose and Expression Normalization for Face Recognition in the Wild," in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 787–796.

[9] P. Kellnhofer, A. Recasens, S. Stent, W. Matusik, and A. Torralba, "Gaze360: Physically unconstrained gaze estimation in the wild," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6912–6921.

[10] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "Mpiigaze: Real-world dataset and deep appearance-based gaze estimation," IEEE transactions on pattern analysis and machine intelligence, vol. 41, no. 1, pp. 162–175, 2017.

[11] N. Ruiz, E. Chong, and J. M. Rehg, "Fine-Grained Head Pose Estimation Without Keypoints," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 2155–215509, 2018.

[12] Z. Cao, Z. Chu, D. Liu, and Yingjie Chen, " A Vector-based Representation to Enhance Head Pose Estimation," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), January 2021, pp. 1188–1197.

[13] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "It's Written All Over Your Face: Full-Face Appearance-Based Gaze Estimation," in Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on. IEEE, 2017, pp. 2299–2308.

[14] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the Continuity of Rotation Representations in Neural Networks," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5738–5746, 2019.