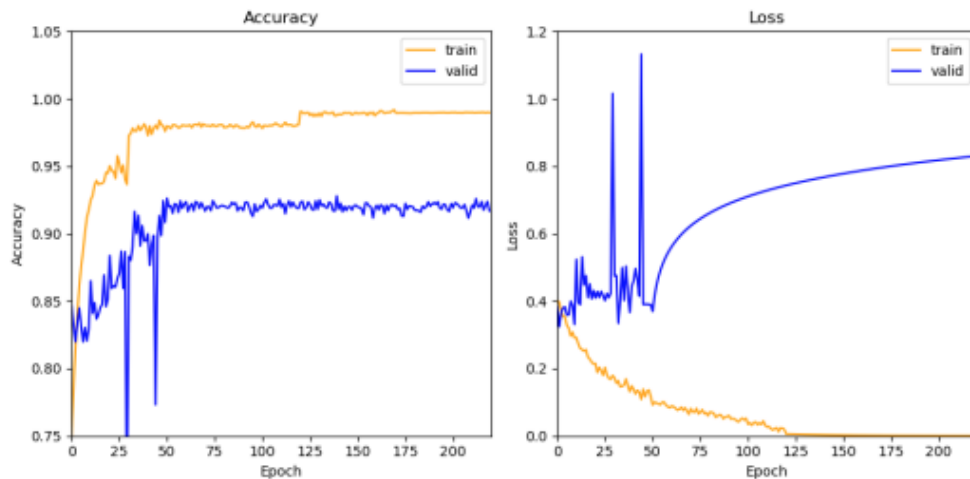


3. You're currently training a neural network to solve the facial detection problem. Below is the loss and accuracy chart you obtained from the training. What is the current issue with the training, and how can this issue be resolved?

Model: ResNet-152

Data: 1000 face detection labeled data



針對 ACC 以及 Loss 兩張圖表，可以得出幾個問題：

1. Overfitting

- 從左圖可以看到，訓練集的準確率非常高（接近 1.0），但驗證集的準確率較低，且波動較大（大約在 0.85 左右）。
- 從右圖可以看到，訓練集的損失值持續下降，但驗證集的損失值卻在一段時間後開始上升，這是典型 Overfitting 現象。

2. 驗證損失值不穩定

- 驗證損失值在訓練初期有大幅度波動，顯示模型可能對驗證集數據的泛化能力不足。

問題原因：

1. 數據量不足：

- 只有 1000 張標註數據，可能無法支撐 ResNet-152 這樣的大型模型，導致模型容易記住訓練數據（過擬合）。

2. 過於複雜的模型：

- ResNet-152 是深層網路，參數量非常大，在小數據集上容易導致 Overfitting。

解決方案：

1. 數據增強 (Data Augmentation)：

- 增加數據多樣性（例如旋轉、翻轉、縮放、色彩變化等），提高模型的泛化能力。

2. 使用更小的模型：

- 考慮使用較小的模型（如 ResNet-18 或 ResNet-34）來避免過度擬合。

3. 正則化方法：

- 使用 Dropout、L2 正則化等方式來減少過擬合。

4. 使用預訓練模型進行遷移學習 (Transfer Learning)：

- 在較大的數據集（如 ImageNet）上進行預訓練，再微調模型以適應

人臉偵測任務。

5. 調整訓練策略：

- 使用 Early Stopping，在驗證集損失開始上升時停止訓練。

6. 增加數據量：

- 收集更多的人臉數據，擴大訓練集的規模。

4. Building a Neural Net

Imagine you're going to implement a deep neural network which detects human faces appear in video frames captured by a surveillance camera in your office. You have created a network architecture includes a brand-new network layer you've implemented by yourself. You now have 100,000 labeled face images collected from some social media. Those images are mostly captured and uploaded with the mobile devices (e.g., selfies) of the social media users. Also, you have 1,000 labeled image frames collected directly through the surveillance camera in your office.

How would you arrange your 101,000 images for model training?

Once you've built your self-implemented neural network. What would you do to sanity check the implementation before you start your massive network training workload?

問題 1：如何安排 101,000 張圖片進行模型訓練？

圖片資料集來源：

1. 100,000 張標註好的社交媒體照片 (多數是自拍，來自移動設備)。
2. 1,000 張來自辦公室監控的標註影像。

根據資料集可以發現有很嚴重的 Data imbalance 狀況，針對這種狀況，可以使用幾種方式處理：

1. 針對 1,000 張來自辦公室監控的標註影像使用資料增強(傳統的電腦視覺方法或是 stable diffusion 用於增加資料數量)。(效果可能較差)
2. 使用加權損失函數 (Weighted Loss Function)，可以針對少數類別的監控資料分配更高的權重。
3. 使用在別的自然影像 dataset 的預訓練模型權重進行遷移學習(Transfer Learning)。
4. 首先訓練社交媒體的資料集，接下來再使用辦公室監控的資料集用於 fine-tune。

問題 2：如何進行 Sanity Check？

1. 檢查 data：
 - 確保 dataset 加載、標註和預處理（如正規化、調整大小）正確。
 - 可視化數據樣本，檢查是否存在對應錯誤（如標籤對齊錯誤）。
2. 測試網路架構：
 - 使用小型數據集（約 10 張圖片）進行快速測試，能確保模型能成功前向傳播，損失值能正常計算，且小型的數據集可以測試是否會快速 overfitting。

3. Learning rate(lr)檢查：

- 使用一部分數據（如驗證集）觀察不同學習率下的收斂情況，確認最佳學習率範圍。

4. 邏輯檢查：

- 訓練模型時印出中間層輸出，確認特徵提取合理性。
- 使用單一樣本或類別測試，觀察是否有顯著的錯誤。