

一、實驗設計與結果

本章節中，根據第三章之研究方法以各項實驗驗證系統之設計，並介紹各實驗之目的與設計、評估方式以及結果與分析，透過以下五章子節進行說明：臉部偵測準確度實驗、臉部偵測執行時間實驗、臉部遮擋辨識實驗、姿勢辨識實驗及影片危險偵測實驗。

另外，本章各項實驗皆於相同硬體環境操作，詳細資訊如下：

- 作業系統：Windows 10
- CPU：Intel(R) Core(TM) i7-10700KF CPU @ 3.80GHz
- 記憶體：128GB
- GPU：NVIDIA GeForce GTX 1660

1.1 臉部偵測準確度實驗

在收集嬰兒臉部資料集時，需針對嬰兒影像擷取出臉部範圍，進而後續之臉部遮擋辨識階段。

為了使本系統擁有較佳的臉部偵測準確性且兼具執行效能，本文透過1.1節及1.2節之實驗，分別進行臉部偵測演算法準確度與執行時間之比較，進而驗證以下設計：先使用 SSD 演算法偵測嬰兒臉部，此方法之召回率雖低，但其準確度很高，故能利用此算法之時間優勢；而若 SSD 演算法找不到嬰兒面部時，則接續使用 RetinaFace 演算法，利用其正確率及準確率皆高之優點進行嬰兒臉部偵測。

1.1.1 實驗目的與設計

本實驗為計算人臉偵測演算法之嬰兒面部擷取準確度，使用??節的嬰兒姿勢資料集共 15416 張影像，分析 RetinaFace **deng_retinaface_2020**、MTCNN **zhang_joint_2016**、SSD **ye_face_2021**及 Haar cascade **goyal_face_2017**等演算法之偵測結果。

1.1.2 實驗評估方式

透過分類標註四項演算法偵測嬰兒臉部之結果影像，計算出各演算法的 accuracy、precision 及 recall。

1.1.3 實驗結果與分析

由表 1.1、表 1.2、表 1.3及表 1.4 分別為 RetinaFace、MTCNN、SSD 及 Haar cascade 之詳細實驗結果。

表 1.1: RetinaFace **deng_retinaface_2020**偵測嬰兒臉部結果

	預測有臉	預測無臉
實際有臉	12925	11
實際無臉	33	2447
Total	12958	2458

表 1.2: MTCNN **zhang_joint_2016**偵測嬰兒臉部結果

	預測有臉	預測無臉
實際有臉	9361	3399
實際無臉	517	2140
Total	9877	5539

而再經計算後，四項演算法之 accuracy、precision 及 recall 值如表 1.5。

故透過本實驗結果可得出選用 RetinaFace 演算法行嬰兒臉部偵測，可擁有較佳的偵測準確度。另外，值得注意的是，雖然 SSD 的 accuracy

表 1.3: SSD ye_face_2021偵測嬰兒臉部結果

	預測有臉	預測無臉
實際有臉	4830	8141
實際無臉	5	2440
Total	4835	10581

表 1.4: Haar cascade goyal_face_2017偵測嬰兒臉部結果

	預測有臉	預測無臉
實際有臉	2882	9546
實際無臉	725	2263
Total	3607	11809

及 recall 很低，但其 precision 達 99.90%，也就是說其將多數影像誤判為無臉，但判斷為有臉的結果幾乎正確。

因此，本研究接續進行1.2節的實驗，希望利用 SSD 這樣的特質。

表 1.5: 人臉偵測準確度結果

演算法	Accuracy	Precision	Recall
RetinaFace	99.71%	99.75%	99.91%
MTCNN	74.60%	94.78%	73.36%
SSD	47.16%	99.90%	37.24%
Haar cascade	33.37%	79.90%	23.19%

1.2 臉部偵測執行時間實驗

本研究進行嬰兒臉部偵測除了考量準確度外，亦希望提升整體系統之執行效率。

1.2.1 實驗目的與設計

本實驗為計算人臉偵測演算法之執行時間，使用??節的嬰兒姿勢資料集共 15416 張影像，分析 RetinaFace **deng_retinaface_2020**、MTCNN **zhang_joint_2016**、SSD **ye_face_2021**及 Haar cascade **goyal_face_2017**等演算法，其臉部偵測時間之結果；再以嬰兒影片進行實驗，比較本文所設計的兩步驟流程及僅使用 RetinaFace 演算法之偵測時間。

1.2.2 實驗評估方式

透過計算四項演算法偵測完整資料集所花費之時間，分析各演算法平均偵測一張影像之執行時間；並比較有無使用本文方法之臉部偵測時間。

1.2.3 實驗結果與分析

RetinaFace、MTCNN、SSD 及 Haar cascade 四項演算法之詳細實驗結果，請見表 1.6。

表 1.6: 人臉偵測執行時間結果

演算法	總花費時間 (15416 張影像)	每張影像平均時間
RetinaFace	5 小時 42 分 2.10 秒	1.33 秒
MTCNN	2 小時 8 分 22.05 秒	0.50 秒
SSD	9 分 17.26 秒	0.04 秒
Haar cascade	18 分 01.78 秒	0.07 秒

故透過本實驗結果可得出使用 SSD 演算法進行嬰兒臉部偵測，將可擁有較佳的偵測速度。而1.1節實驗中，準確度最高的 RetinaFace 其平均偵測一張影像需 1.33 秒，為 SSD 的 33.25 倍。

而使用本文設計之兩步驟偵測流程及僅使用 RetinaFace 演算法之詳細實驗結果，請見表 1.7。

表 1.7: 使用本文方法之人臉偵測執行時間結果

方法	總花費時間 (1594 張影像)	每張影像平均時間
本文方法	17 分 51.21 秒	0.67 秒
僅使用 RetinaFace	32 分 2.69 秒	1.21 秒

因此，總結1.1節與1.2節之實驗結果，驗證本系統先使用 SSD 演算法偵測嬰兒臉部，未如期找到目標時，則改以 RetinaFace 演算法偵測，達成兼具準確性及執行效率之系統目標。

1.3 臉部遮擋辨識實驗

本研究中，利用深度學習技術辨識嬰兒臉部是否遭非奶嘴之異物遮蔽，進而判斷嬰兒是否處於危險情境中。

1.3.1 實驗目的與設計

本實驗為訓練針對嬰兒臉部遮擋辨識之模型，以 ResNet50 he_deep_2016 訓練??節的嬰兒臉部資料集，並透過驗證集進行模型驗證。

程式實作中，網路訓練回合數為 20，設定影像資料大小為 224x224，包含三個類別（臉部無遮擋之安全狀態、使用奶嘴及面部遭異物遮蔽之警示狀態），且透過 data augmentation 技術生成訓練及測試資料，輸出層使用 softmax 作為激發函數，並使用 Adam 作為 optimizer 且將學習率設為 0.000001 以進行微調。

1.3.2 實驗結果分析

模型最終訓練準確率達 98.06%，而測試準確率達 99.43%，詳細訓練結果請見圖 1.1。

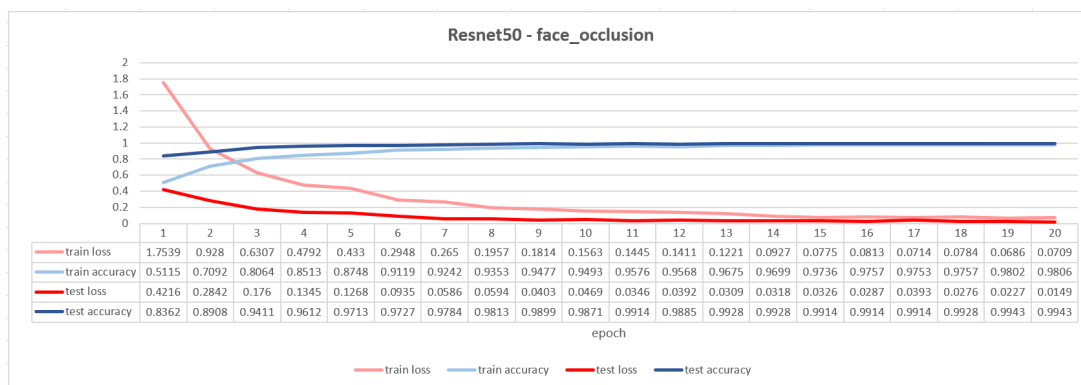


圖 1.1: 臉部辨識訓練及測試結果

我們使用 342 張之驗證集影像進行模型驗證，所有影像皆辨識正確。此模型之混淆矩陣如表 1.8，表中數字為各類影像之張數及比例，並於最右行展示各類別召回率。

表 1.8: 臉部遮擋辨識模型之混淆矩陣（單位：張 / 百分比）

		預測類別			召回率
		安全	奶嘴	警示	
實際類別	安全	120 (100.00)	0 (0.00)	0 (0.00)	100.00%
	奶嘴	0 (0.00)	115 (100.00)	0 (0.00)	100.00%
	警示	0 (0.00)	0 (0.00)	107 (100.00)	100.00%

1.4 姿勢辨識實驗

本研究中，利用深度學習技術辨識嬰兒基礎姿勢，進而判斷嬰兒是否處於危險情境中。

1.4.1 實驗目的與設計

本實驗為訓練針對嬰兒姿勢辨識之模型，以 ResNet50 he_deep_2016 訓練??節的嬰兒姿勢資料集，並透過驗證集進行模型驗證。

程式實作中，網路訓練回合數為 20，設定影像資料大小為 224x224，包含四個類別（正躺、趴躺、坐姿及站立），且透過 data augmentation 技術生成訓練及測試資料，輸出層使用 softmax 作為激發函數，並使用 Adam 作為 optimizer 且將學習率設為 0.000001 以進行微調。

1.4.2 實驗結果分析

模型最終訓練準確率達 99.45%，而測試準確率達 99.71%，詳細訓練結果請見圖 1.2。



圖 1.2: 姿勢辨識訓練及測試結果

我們使用 744 張之驗證集進行模型驗證，包含了五張類別辨識錯誤的影像，其中三張將坐姿誤判為趴躺姿勢，推測原因為嬰兒雖呈現坐姿，但上半身貼近其腿部（如圖 1.3），而導致誤判。此模型之混淆矩陣如

表 1.9，表中數字為各類影像之張數及比例，並於最右行展示各類別召回率。



圖 1.3: 姿勢辨識錯誤之影像：坐姿誤判為趴躺

表 1.9: 姿勢辨識模型之混淆矩陣（單位：張 / 百分比）

		預測類別				召回率
		正躺	趴躺	坐姿	站立	
實際類別	正躺	164 (100.00)	0 (0.00)	0 (0.00)	0 (0.00)	100.00%
	趴躺	1 (0.52)	191 (99.48)	0 (0.00)	0 (0.00)	99.48%
	坐姿	0 (0.00)	3 (1.50)	196 (98.00)	1 (0.50)	98.00%
	站立	0 (0.00)	0 (0.00)	0 (0.00)	192 (100.00)	100.00%

1.5 影片危險偵測實驗

本研究基於嬰兒影像進行臉部遮擋及姿勢辨識，透過讀取嬰兒影片達成危險監測之目標。

1.5.1 實驗目的與設計

本實驗為驗證此系統能基於嬰兒影像進行危險監測，利用網路之真實嬰兒影片，包含不同之拍攝視角、嬰兒樣貌及狀態等，實驗臉部遮擋辨識模型與姿勢辨識模型之準確性。

1.5.2 實驗評估方式

透過輸出每幀影像之臉部遮擋及姿勢辨識結果，計算其 accuracy、precision 及 recall，以驗證此二模型得以應用在監測嬰兒危險情境。

1.5.3 實驗結果分析

本實驗使用之影片，包含嬰兒許多的不同情境，如：清醒與否、不同衣著、穿戴帽子與否及相異背景環境等，進行其臉部遮擋（無遮蔽、使用奶嘴及遭異物遮擋）與姿勢（正躺、趴躺、坐姿及站立）之危險辨識。

首先，姿勢辨識的部分，包含了多張誤判為趴躺姿勢的影像，推測原因為嬰兒身體遭棉被遮擋（如圖 1.4），而只拍攝到露出的嬰兒臉部，故造成姿勢辨識錯誤。



圖 1.4: 姿勢辨識錯誤之影像：正躺誤判為趴躺

其次，臉部遮擋辨識的部分，會先刪去嬰兒臉部未被偵測之影像（如圖 1.5），而後判斷有多張影像類別應為嬰兒正在使用奶嘴或安全狀態，但誤判為遭異物遮蔽之警示狀態，推測原因為影像中之奶嘴或嬰兒臉部遭手部等遮擋（如圖 1.6），而誤判類別。



圖 1.5: 未偵測嬰兒臉部之影像

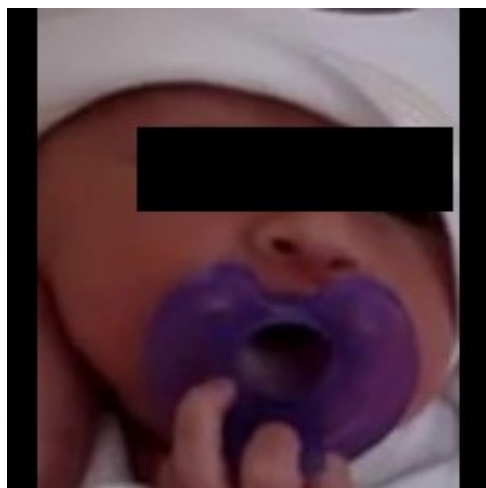


圖 1.6: 臉部遮擋誤判之為警示狀態

兩部分之混淆矩陣如表 1.10及表 1.11，表中數字為各類影像之張數及比例，並於最右行展示各類別召回率。

表 1.10: 實驗影片姿勢辨識之混淆矩陣（單位：張 / 百分比）

		預測類別				召回率
		正躺	趴躺	坐姿	站立	
實際類別	正躺	3223 (90.87)	324 (9.13)	0 (0.00)	0 (0.00)	90.87%
	趴躺	0 (0.00)	161 (100.00)	0 (0.00)	0 (0.00)	100.00%
	坐姿	0 (0.00)	0 (0.00)	214 (100.00)	0 (0.00)	100.00%
	站立	4 (1.87)	1 (0.47)	0 (0.00)	209 (97.66)	97.66%

表 1.11: 實驗影片臉部遮擋辨識之混淆矩陣（單位：張 / 百分比）

		預測類別			召回率
		安全	奶嘴	警示	
實際類別	安全	1429 (85.83)	52 (3.12)	184 (11.05)	85.83%
	奶嘴	17 (0.84)	1615 (80.23)	381 (18.93)	80.23%
	警示	40 (11.90)	0 (0.00)	296 (88.10)	88.10%