

中華民國第 60 屆中小學科學展覽會 作品說明書

高級中等學校組 電腦與資訊學科

探究精神獎

052503

以人工智慧分析眼白影像和黃疸的關聯性

學校名稱：臺中市私立曉明女子高級中學

作者： 高二 陳郁媗 高二 曾翊甯 高二 陳采昕	指導老師： 黃齡慧
---	------------------

關鍵詞：人工智慧、黃疸、圖像辨識

摘要

- 研究動機：黃疸為常見肝臟疾病表徵，對於偏鄉缺乏醫療的環境，民眾與醫師需有方便的篩檢設備。
- 研究目的：利用眼部照片以多元迴歸方程式、深度學習，建構眼白顏色與黃疸程度的預測模型。
- 研究設備及器材：與中國醫藥大學合作，取得 354 張眼部照片與膽紅素值，利用 Python 撰寫程式進行人工智慧運算，及利用 Excel 的迴歸工具進行多元迴歸方程式推導。
- 研究過程：以程式擷取眼白像素，進行多元迴歸分析並推導出預測方程式，同時進行深度學習，計算預測率，並以 APP 成現。
- 研究結果：多元迴歸分析得預測率約為 45.5%~57.8%，深度學習則可高達 88.9%~99%。
- 結論：我們成功以深度學習根據眼部照片預測黃疸的嚴重程度。

壹、研究動機

黃疸為許多代謝相關疾病的症狀，最典型常見的體徵為皮膚及眼白變黃。但由於部分人原膚色就偏黃或誤認變黃為曬黑等等，使得疏忽了疾病與器官衰竭，而發現、診斷出黃疸或器官病變時病情已加劇。故希望透過此研究更加了解黃疸，協助民眾盡早察覺肝臟病症、快速就醫。

貳、研究目的

偏鄉地區醫療資源分配不均的狀況日益加重，應相對應找尋解決改善方法，以行動裝置運用人工智慧分析眼白影像與黃疸之關係能夠協助醫療機構、診所做篩檢。另外，因為人工智慧建立在電腦計算，而電腦現今的運算速度及效率已超越人類，因此以人工智慧協助篩檢能大幅減少程序與人力，提升效率。

參、研究設備及器材

一、標準色卡

以標準色卡上的白色陶瓷為基準白色，照相時放在臉旁導引相機設定為最亮值，使照片亮度誤差較小。

二、相機

使用相機或手機紀錄正常人與黃疸病患的照片。

三、電腦及程式軟體

(一) 多元線性迴歸分析部分

以 Python3.7 編寫程式碼以判斷病患眼白顏色偏黃程度，並在 Spyder 編譯器下的環境處理經白平衡後的照片。處理過後的數據將會記錄於 Excel 中，並

用 Excel 內建的工具執行多元迴歸分析，以取得最佳函數解，並將其轉成 Excel 檔存取，方便未來將其數據使用。

（二）深度學習部分

以 Python3.7 編寫程式，篩選淨化原始資料後，利用影像處理軟體擷取眼睛的圖案，在套用相關數值及圖形運算模組（numpy、pandas、matplotlib、opencv）進行前處理。在強化效能部分先運用資料增強的方式將比例較少的資料增加，再使用 EfficientNet 套件執行轉移學習。最後再訓練部分是運行在 Tensorflow 的框架上。

肆、研究過程或方法

一、研究方法

（一）生理觀察

黃疸非一種單一疾病，而是某些疾病的症狀表現。黃疸的產生原因為身體血液內膽紅素（bilirubin）過多而造成皮膚、眼白、黏膜處等顏色偏黃。膽紅素是膽汁的主要色素，為紅血球破壞後細胞內的血色素所製造。其中會先產生「未結合型膽紅素（Unconjugated bilirubin）」，又或稱「間接型膽紅素（Indirect Bilirubin）」，接著隨血液流進肝臟代謝，並在肝細胞中與葡萄糖醛酸結合，形成「結合型膽紅素（Conjugated bilirubin）」，或稱為「直接型膽紅素（Direct Bilirubin）」。生成的結合型膽紅素會再隨著血液流往腎臟或隨著膽汁進入消化道，最後再由糞便和尿液排出體外。

若是分解的紅血球太多、肝臟代謝功能衰敗、或是膽汁移至胃腸的路徑受到阻塞皆有可能造成黃疸的發生。溶血性疾病會增加被破壞的紅血球數量；病毒、藥物則可能造成肝功能失常；膽結石或膽管癌則會使膽管阻塞，膽紅素便無法通行。由以上可知造成黃疸的原因有很多種，因而一旦發現有黃疸的症狀宜盡快就醫檢查。

較典型的黃疸症狀為：皮膚、眼白顏色變黃、尿液顏色如濃茶或可樂色、大便變成灰白色、皮膚變癢、或者以上症狀伴隨著發燒、畏寒、腹痛、類感冒、體重下降等症狀。其中又以判斷眼白的顏色變化較為準確，因為皮膚變黃而眼白不黃有可能是食用過多 β 胡蘿蔔素所導致。每個人的膚色可能因日曬或種族等原因有不同的顏色，但所有人的眼睛鞏膜顏色皆為白色，較不易導致主觀判斷錯誤。本實驗由拍照來判斷病患是否有黃疸症狀，而每位病患皆會拿著一張標準色卡以確認相片最亮點較為一致。標準色卡的白色為陶瓷所燒製而成，較不會隨著時間改變而導致顏色有誤差。第二段的電腦色彩判讀將會詳細解釋如何使用這些照片來判定黃疸症狀。

（二）電腦色彩判讀

三原色光模式（RGB color model），又稱 RGB 顏色模型或紅綠藍顏色模型，是一種加色模型，主要功能是將紅（Red）、綠（Green）、藍（Blue）三原色的色光以不同的比例相加以合成產生各種色彩光。由於人類視覺感色細胞的感光構造，紅綠藍成為三原色合成色彩的基礎。目前在電腦硬體中採取每一象素用 24bit（位元）表示的方法，三種原色光各分到 8bit，因此每一種原色的強度依照 8bit 的最高值 2 的 8 次方分為 256 個值。三種顏色都可分為 0~255 單位等級的原色強度，並且可用這種方法組合成 16777216 種顏色。

在此實驗中，我們將會搜集正常人及黃疸患者眼白顏色的照片。各眼白顏色會以標準色卡做為對照，以便之後以標準色卡的顏色為基準做白平衡等色彩的校正，使各項數據變得更加準確且客觀。接著再藉由程式分析照片中的眼白顏色，歸納出此眼白中 RGB 所各占的比例。實驗中會以同一張照片進行多次分析，因為眼白各區域可能因拍攝角度等原因造成分析出的顏色比例有些不同。若只取照片中任意一點，便可能因眼白反光、有陰影遮罩等因素造成數據不夠客觀，進而影響最終分析結果。

(三) 人工智慧

1. 多元線性迴歸分析

多元線性迴歸分析 (Multiple Regression Analysis) 為探各討變數之關係的迴歸模型，一般模型為： $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p + e$ 。

X：自變數（分析對象）、Y：應變數（對應結果）、 β ：權重、e：偏差值。

在此研究程式中，我們將 R、G、B 三數值設為 X 變數（R 為 X_1 、G 為 X_2 ……依此類推），並將膽紅素值設為 Y 變數。 β_0 為 Y 截距（起始值）， $\beta_1 \sim \beta_p$ 則為未知且須估計的迴歸係數。我們以最常見的分析方法：最小平方法（Least squares）（尋找數據的最佳函數，使這些求得的數據與實際數據之間誤差的平方和為最小之方法）求得 β 值，並運用 Excel 多元迴歸分析的預設工具達到此目的。e 為數值與起始值間的差距。所以在此研究中，我們將尋求多元迴歸分析的目標函數或是損失函數（loss function），也就是找到一組最佳的參數（神經網路中的權重和偏值），使得函數值最小。

2. 人工神經網路

人工神經網路 (Artificial Neural Networks, ANN)，又稱為類神經網路，啟發於生物神經網路，模擬生物神經網路的運作方式，使電腦具備學習與記憶能力，進而做出事物間的連結、做出判斷。類神經網路運用大量人工神經元建構機器學習 (Machine Learning) 的計算模型，建立神經元間的連結、彼此影響，而透過計算的過程及結果，對模型本身做調整及改良，達成學習的效果。

類神經網路主要分為三個階層：輸入層（Input layer）、隱藏層（Hidden layer）、輸出層（Output layer）。輸入層為眾多接收刺激訊息的神經元（neuron），能將接收到的訊息傳至下一層。隱藏層可為單層或多層，由多個神經元組成，夾在輸入層及輸出層之間。隱藏層及各隱藏層中的神經元數目是影響神經網路學習效能的重要因素。在隱藏層中，訊息會經過計算、轉換、分析、整理等處理，而一層一層傳遞至輸出層。接收到經過層層神經元處理後的訊息，輸出層中反應最為強烈（數值最高）的神經元代表此神經網路的辨識結果，作為最後的輸出項目。

通常為了決定傳遞至下一個神經元的訊息，我們會賦予它一個激活函數（activation function），讓刺激微弱（數值太小）到可忽略的神經元輸出 0（沒有任何影響力）；讓刺激強烈（數值很大）的神經元輸出 1；其他設於 0 到 1 之間。

機器學習（machine learning）是人工智慧的一個子集，自動運用電腦上各種演算法與統計模型執行任務以取代繁雜的指令。機器學習組合的數學模型建立在訓練範例（數據、資料等）上，用來預測結果。主要分為監督式學習（supervised learning）和非監督式學習（unsupervised learning）。監督式學習必須事先給定標籤（label，分類的結果），由訓練資料中得出一個規則，進而預測新的目標對象；非監督式學習不需要事先給定標籤，能自動對輸入的資料進行分類、分群、分析，得出規律。此研究使用監督式學習。

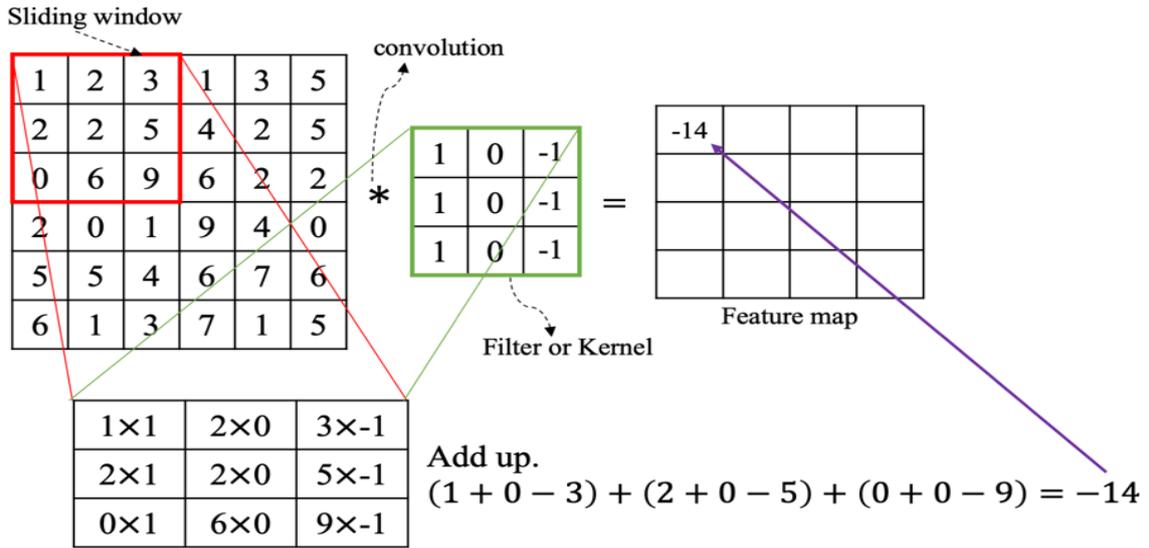
3.卷積神經網路

卷積神經網路（Convolutional Neural Network, CNN）專門在做圖像識別，對於照片或影片的處理是最適合的神經網路。此研究的核心在於圖像識別，故使用卷積神經網路。

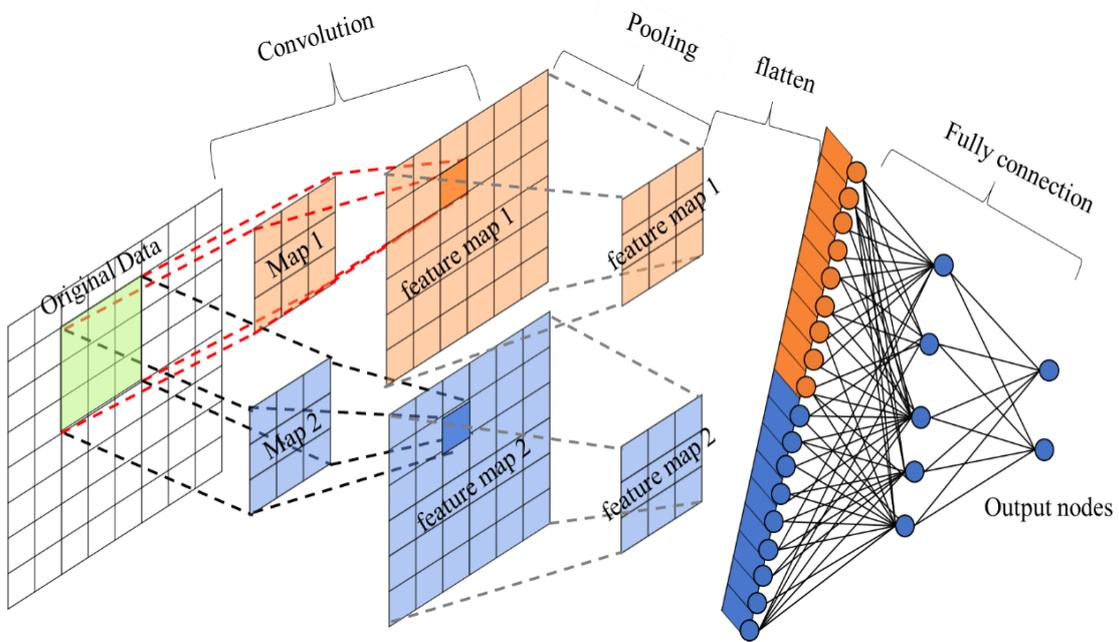
卷積神經網路包括兩種層：卷積層（convolution layer）及池化層（pooling layer）。辨識照片時，通常會設定一個過濾器（filter）去看照片中的某個特徵，不同過濾器擁有識別不同目標特徵的功能。在卷積層中，過濾器會掃描圖片，將圖片中不同區塊的特徵強度記錄下來。對於同一張照片，可以使用多種過濾器，以觀察多種特徵。在池化層中，會將特徵圖（feature map）分為多個區域，再選出特徵強度最強的數值（可以最大化、平均化、最小化等）代表此區域，將特徵圖降維。卷積神經網路可以做多次的卷積和池化，最後會接上全連接層（fully-connected layer）後輸出。以下會做更詳細的卷積運算流程。

在圖像辨識前，要先對影像（以此研究而言就是眼睛照片）做預處理程序（preprocessing），讓電腦更有效的學習。空間濾波器（spatial filtering）是預處理程序的一種技巧，運用卷積運算使某種特徵更加突出，使用不同濾波器（filter）會得到不同效果。濾波器又稱為遮罩（mask）或核心（kernel）。

接下來說明卷積運算。以二維向量為例，也就是圖片處理時運用到的流程。設有兩不同大小之矩陣，小矩陣為遮罩，大矩陣為預處理的圖片。參考下圖。首先，將小矩陣置於大矩陣左上角進行乘積，對應到的區域中的數值相乘，再將相乘後的數值相加得到第一個卷積結果。接下來，將遮罩向右移動一單位（stride=1），再進行乘積，得到的結果輸入第一個結果的右一單位位置。一行完成卷積後再往下一單位，並從最左方開始做乘積，結果輸入第一個結果的下一單位位置。卷積運算完的結果就是新的結果矩陣。若欲得到與原圖大小相同的結果矩陣，那會欲先在原向量周圍補上 0（zero-padding）。



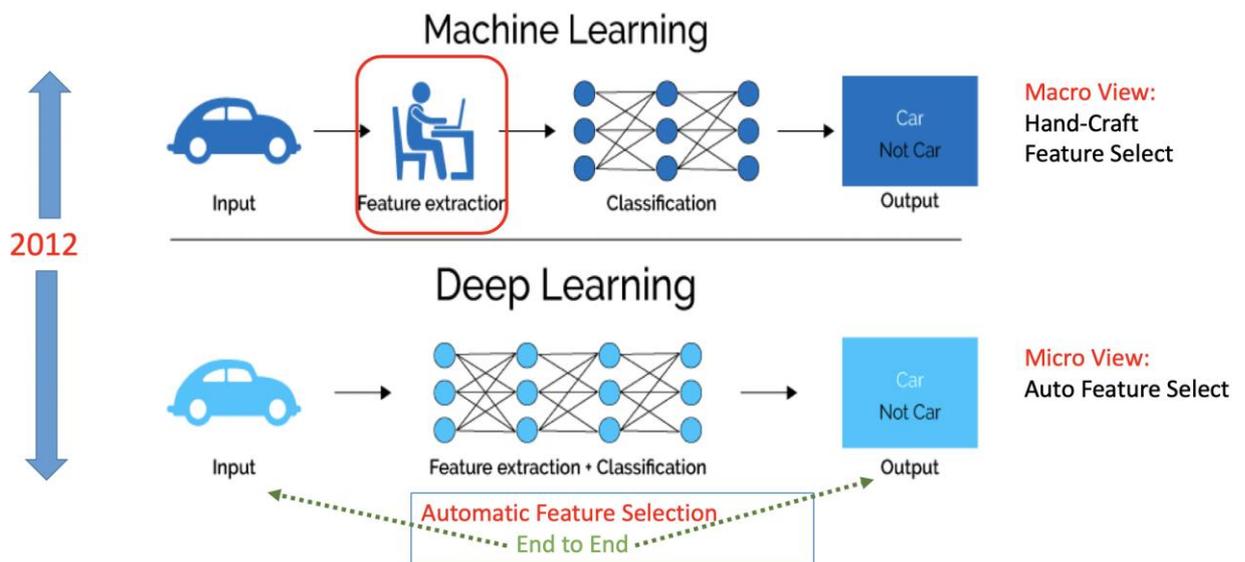
Created by brilliantcode.net



4. 深度學習

深度學習（deep learning）是表徵學習的演算法，將其與其他演算法區別的特徵是能夠做非監督式學習（unsupervised learning）或半監督式學習（semi-supervised learning），以及自行選取特徵取代人工選取。

在 2012 年以前，透過機器學習做人工智慧的分析，必須以人工方式給予特徵值，但是這種方式取得的特徵值數量有限，且既沒有效率又耗時。在 2012 年以後因為 GPU 的導入，進入了深度學習的時代，不再需要人工給予標籤，神經網路就能自行截取特徵值並將其分類、預測結果，達到「端到端（end to end）」的學習。



5. 訓練深度學習模型

訓練模型時，會將資料庫分為三群：訓練集（training）、驗證集（validation）、測試集（testing）。模型會運用訓練集進行學習，針對這些圖片調整參數，使其能夠辨識這些圖片。驗證集是提供模型去驗證，調適其自身結構，如神經元個數等。最後，模型利用測試集檢驗辨識、分類效果。

6. 應用

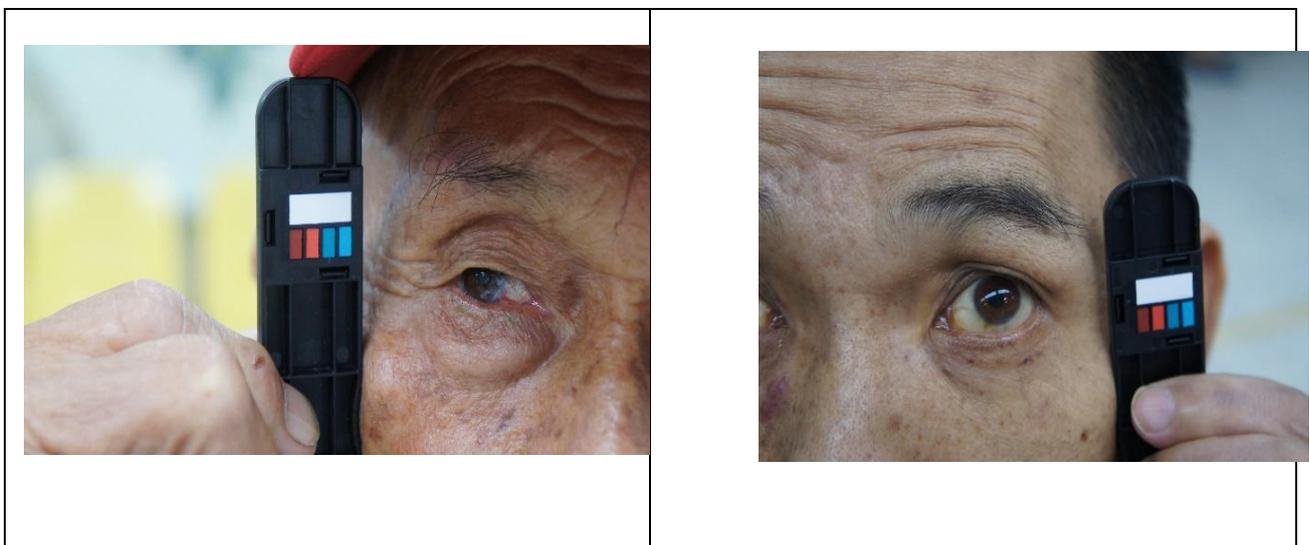
目前人工智慧技術主要應用於下列幾項大點：

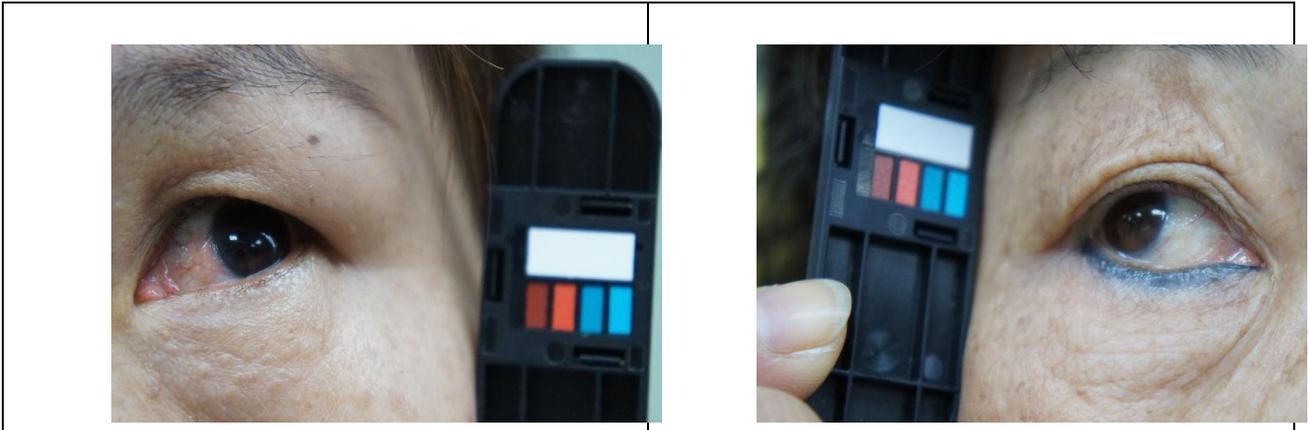
- (1) 健康照護與醫療應用
- (2) 智慧製造
- (3) 商業分析
- (4) 網路安全應用
- (5) 教育與研究
- (6) 電腦視覺
- (7) 語音工程
- (8) 自然語言處理
- (9) 移動與控制

二、研究過程

(一) 收集資料

我們和中國醫藥大學附設醫院合作，經病人同意後，分別拍攝病人的左右眼，取得總計 354 張照片，並收集病人的實際膽紅素濃度資料。以下研究過程說明，運用所收集的資料做兩種分析。如下圖：





(二) 多元迴歸分析

1. 取得眼白像素 RGB

我們在 python3.7 環境下以 Spyder 3 編譯器撰寫程式，輸入實際膽紅素數值後程式會自動開啟照片，接著便人工點取圖片中眼白的部分，以取得所點選到的像素之 RGB。依照其膽紅素濃度不同，點的數量也會不同。我們將照片分為三類別，膽紅素濃度 0~1.2mg/dL 點 20 個像素、1.3~3.5mg/dL 點 40 個像素、3.6mg/dL 以上點 50 個像素，並盡量隨機分布各點，適時避開反光、血絲、陰影、眼黑邊緣等會影響研究結果的部分。

2. 運用 Excel 進行多元迴歸分析

我們藉由程式分析各張圖片的各點 R、G、B 所佔強度比例之數據，總計 4320 筆資料，並彙整成 Excel 檔案以利分析。在 Excel 程式中，需先點選〔檔案〕>〔選項〕>〔自訂功能區〕中將右側〔開發人員〕模式打勾。回到工具列點選〔開發人員〕並開啟〔增益集〕將〔分析工具箱〕打勾。回到工具列〔資料〕一欄，右側會出現〔資料分析〕，點選後選取「迴歸」選項，並依其要求輸入對應數值，開始進程式運算。將膽紅素濃度設為 Y 值（應變數），接著再點選 R、G、B 數值設為其餘對應值，讓 Excel 跑完程式。

3. 得到回歸模型

此為 Excel 程式執行後顯示的模型：

	A	B	C	D	E	F	G	H	I
1	摘要輸出								
2									
3	迴歸統計								
4	R 的倍數	0.4511432							
5	R 平方	0.20353019							
6	調整的 R 平方	0.20297772							
7	標準誤	1.37313483							
8	觀察值個數	4329							
9									
10	ANOVA								
11		自由度	SS	MS	F	顯著值			
12	迴歸	3	2083.87661	694.625535	368.404032	4.502E-213			
13	殘差	4325	8154.78436	1.88549927					
14	總和	4328	10238.661						
15									
16		係數	標準誤	t 統計	P-值	下限 95%	上限 95%	下限 95.0%	上限 95.0%
17	截距	1.79009078	0.10775769	16.6121858	4.0047E-60	1.57883046	2.00135109	1.57883046	2.00135109
18	R	-0.00904534	0.00193041	-4.68571426	2.8756E-06	-0.01282993	-0.00526075	-0.01282993	-0.00526075
19	G	0.06795248	0.00376582	18.044533	3.0956E-70	0.06056954	0.07533542	0.06056954	0.07533542
20	B	-0.06244712	0.00236849	-26.3657526	3.406E-142	-0.06709059	-0.05780366	-0.06709059	-0.05780366

16 到 20 欄的數據中，截距即為 β_0 ，同行下方三欄則各為 R、G、B 的對應係數，也就是 β_1 、 β_2 、 β_3 。表格其他數據並非此研究之討論對象，因此不加以解釋。依所得係數可得出公式為： $Y = 1.79009078 + (-0.0090453) X_1 + (0.06795248) X_2 + (-0.0624471) X_3$

將此回歸模型代回 Excel 工具列處的公式欄位計算，即可將所得數值進行後續預測。

4. 將預測結果和實際資料分群

將預測結果和實際資料分群。第一種分為三群：膽紅素濃度 0~1.2 標記為 1（第一群），濃度 1.3~3.5 標記為 2（第二群），濃度 3.6 以上標記為 3（第三群）。第二種分為兩群：濃度 0~1.2 標記為 1（第一群），濃度 1.3 以上標記為 2（第二群）。將各筆資料的預測值與實際值相減得到差值。例如，第一種各筆預測值和對應的實際值相減，第二種以此類推。若差值等於 0（預測值和

實際值相等，表示預測成功）則標記為 0，若大於 0（預測值和實際值不相等，表示預測不成功）則標記為 1。觀察預測結果和實際資料的關係。

5. 驗證回歸模型

將此程式碼帶回各項數值中，依序輸入 β_0 （截距）、 β_1 （R 的對應係數）、 X_1 （R 的數值）、 β_2 （G 的對應係數）、 X_2 （G 的數值）、 β_3 （B 的對應係數）、 X_3 （B 的數值），最後求出 Y（膽紅素濃度，也就是結果）。計算出各個相對應結果後，將計算結果分群（分群方式同上一點分為三群），最後求原始資料與預測結果的分類區塊之交集，得知此函數之準確率。

（三）深度學習

以下逐步說明程式內容。（因為篇幅過長，將省略部分基本程式設計程序。）

1. 匯入相關模組

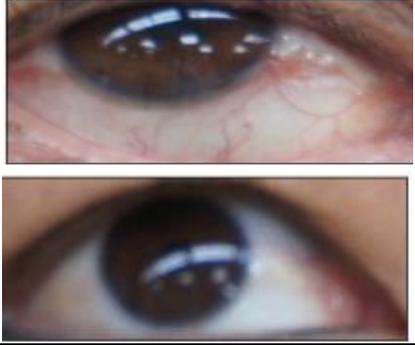
匯入 os（檔案建置與開啟）、numpy（矩陣計算）、pandas（整理資料表格與前處理）、matplotlib.pyplot（繪製圖形）、opencv（影像前處理）等模組。

2. 資料匯入與處理

從 354 張照片篩選解析度佳者，並框選出眼睛部分，使其為較易後續處理的形式。最後選出 113 張照片作為本次深度學習的樣本。

3. 歸類照片類別

將照片歸類成三個類別：膽紅素濃度 0~1.2（51 張）、1.3~3.5（30 張）、3.6 以上（9 張），並控制圖片大小、宣告照片的標題、將指定的影像放入呼叫的畫圖位置等。

類別	數值	影像圖片
1	0~1.2	
2	1.3~3.5	
3	3.6 以上	

4. 影像前處理 (preprocessing)

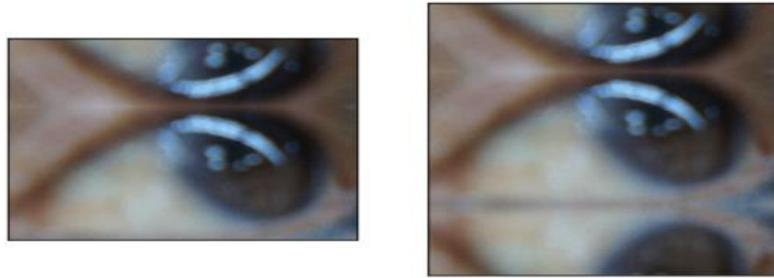
(1) Reflection 鏡射

a. 裁切 (cropping)

將每張照片調整成相同大小。

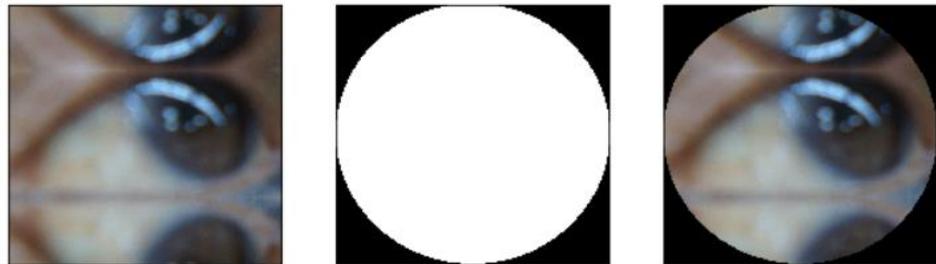
b. 鏡射補值

以長邊為直徑，短邊以鏡射方式補值（以短邊為對稱軸鏡射），直到長寬等長。如下右圖為鏡射補值後影像：



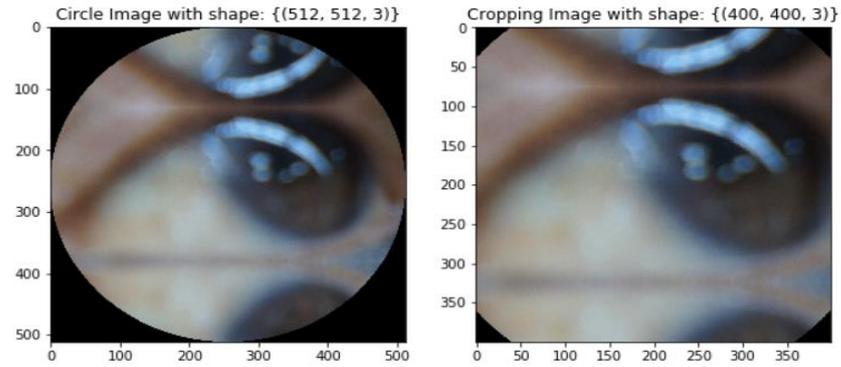
c. 將照片裁切成圓形

以邊為直徑做一相切圓（正方形的內切圓），切除圓以外的影像，只保留圓內的影像做處理。裁減完後確保每一張照片為同一大小，將每張照片設為一特定大小。如下圖：



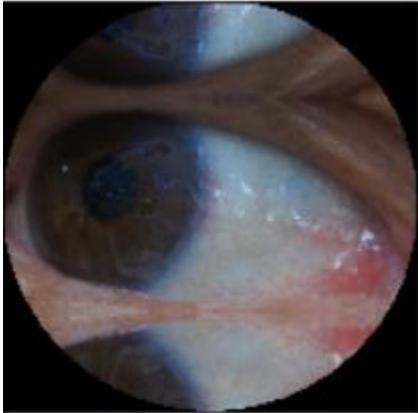
(2) 增加影像面積，減少不必要的特徵

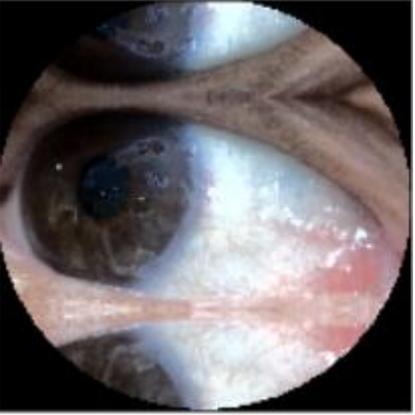
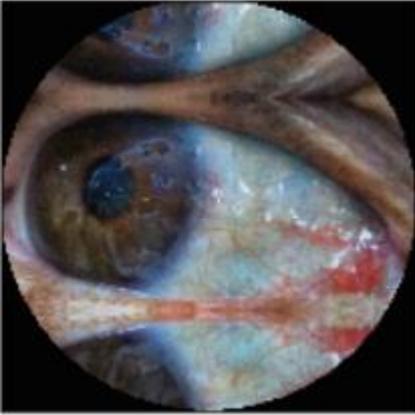
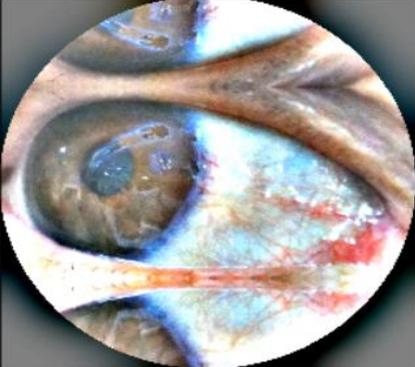
將調整為同一大小的影像再次做裁切。以中心點取一特定值的正方形，只保留正方形內的影像，減少不必要的特徵。如下圖：



(3) Image Processing 影像處理

一般在 CNN 中，運用多種濾波器，擷取各種特徵值，以方便做類神經網路運算，類似傳統影像處理人工選擇濾波器，例如直方圖 Histogram、直方圖均勻化 Clahe (Adaptive Histogram)、高斯 Guassian 等濾鏡。此研究運用深度學習，故不需人工選擇濾鏡，而是 CNN 會自行產生各種未知的濾鏡，可能多至上千種、上萬種。底下列舉幾種濾鏡：

種類	圖片
原始	

直方圖 (Histogram)	
直方圖均勻化 Clahe (Adaptive Histogram)	
高斯 Gaussian	

5. 去除反光

因本研究的目的是偵測眼白影像與黃疸之關係，為避免眼球反光的白色部分可能對深度學習造成干擾，我們利用影像修圖軟體 Photoimpact 的仿製功能將眼睛反光的的部分去掉。如下表：

原始圖形	將眼球反光部分去除
	

6. 增強預測成效

(1) 資料增強 (data augmentation)

將影像水平平移、垂直平移、 0° ~ 180° 任一角度旋轉、任意水平翻轉、任意垂直翻轉、空隙補常數、設定常數值為 0，達到增加資料量的效果。

(2) 轉移學習 (transfer learning)

套用已訓練好的模型，修改部分使其符合此研究。在此我們使用 EfficientNetB5 進行轉移學習。

7. 開始訓練

共進行二十個回合的訓練，每個回合總共重複 33 次的調整權重及參數，底下列出每一回合的正確率 (accuracy)、錯誤率 (loss)、驗證正確率 (validation_accuracy) 及驗證錯誤率 (validation_loss)，正確率可高達 99%，驗證正確率也有 89%。

Epoch 1/20

33/33 [=====] - 386s 12s/step - loss: 1.1176 - accuracy: 0.5954 - val_loss: 1.5267 - val_accuracy: 0.3333

Epoch 2/20

33/33 [=====] - 347s 11s/step - loss: 0.4390 - accuracy: 0.8550 - val_loss: 1.2188 - val_accuracy: 0.4444

Epoch 3/20

33/33 [=====] - 348s 11s/step - loss: 0.2361 - accuracy: 0.9466 - val_loss: 0.8976 - val_accuracy: 0.5556

Epoch 4/20

33/33 [=====] - 347s 11s/step - loss: 0.2378 - accuracy: 0.9160 - val_loss: 0.6209 - val_accuracy: 0.7778

Epoch 5/20

33/33 [=====] - 346s 10s/step - loss: 0.1644 - accuracy: 0.9470 - val_loss: 0.4818 - val_accuracy: 0.8889

Epoch 6/20

33/33 [=====] - 340s 10s/step - loss: 0.1137 - accuracy: 0.9846 - val_loss: 0.7502 - val_accuracy: 0.7778

Epoch 7/20

33/33 [=====] - 343s 10s/step - loss: 0.1669 - accuracy: 0.9084 - val_loss: 0.4879 - val_accuracy: 0.7778

Epoch 8/20

33/33 [=====] - 211s 6s/step - loss: 0.0955 - accuracy: 0.9847 - val_loss: 0.7219 - val_accuracy: 0.6667

Epoch 9/20

33/33 [=====] - 199s 6s/step - loss: 0.1342 - accuracy: 0.9545 - val_loss: 0.7053 - val_accuracy: 0.5556

Epoch 10/20

33/33 [=====] - 197s 6s/step - loss: 0.0748 - accuracy: 0.9769 - val_loss: 0.9278 - val_accuracy: 0.5556

Epoch 11/20

33/33 [=====] - 182s 6s/step - loss: 0.0519 - accuracy: 0.9847 - val_loss: 0.5877 - val_accuracy: 0.6667

Epoch 12/20

33/33 [=====] - 183s 6s/step - loss: 0.0342 - accuracy: 0.9848 - val_loss: 0.4335 - val_accuracy: 0.6667

Epoch 13/20

33/33 [=====] - 183s 6s/step - loss: 0.1052 - accuracy: 0.9695 - val_loss: 0.5858 - val_accuracy: 0.7778

Epoch 14/20

33/33 [=====] - 183s 6s/step - loss: 0.1746 - accuracy: 0.9313 - val_loss: 0.5760 - val_accuracy: 0.6667

Epoch 15/20

33/33 [=====] - 182s 6s/step - loss: 0.1002 - accuracy: 0.9692 - val_loss: 0.4716 - val_accuracy: 0.7778

Epoch 16/20

33/33 [=====] - 184s 6s/step - loss: 0.0261 - accuracy: 1.0000 - val_loss: 0.3577 - val_accuracy: 0.8889

Epoch 17/20

33/33 [=====] - 184s 6s/step - loss: 0.0425 -
accuracy: 0.9924 - val_loss: 0.3091 - val_accuracy: 0.8889

Epoch 18/20

33/33 [=====] - 184s 6s/step - loss: 0.0697 -
accuracy: 0.9771 - val_loss: 0.4447 - val_accuracy: 0.7778

Epoch 19/20

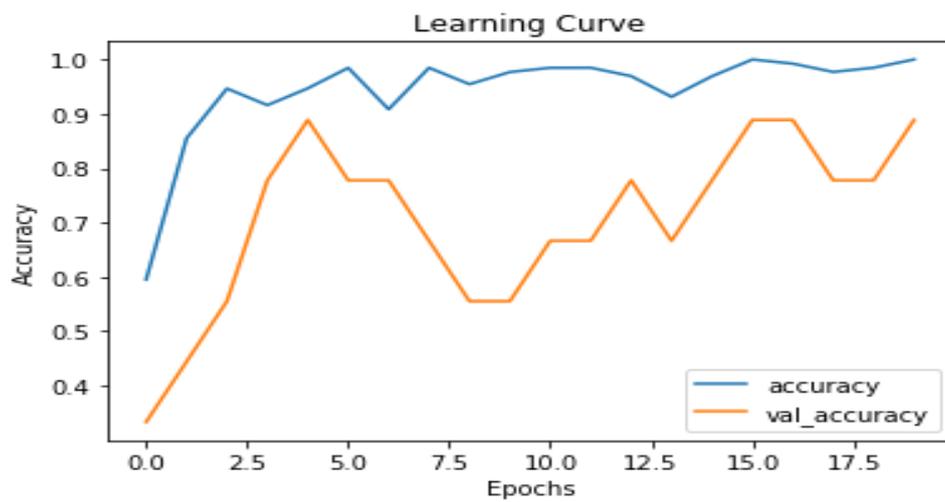
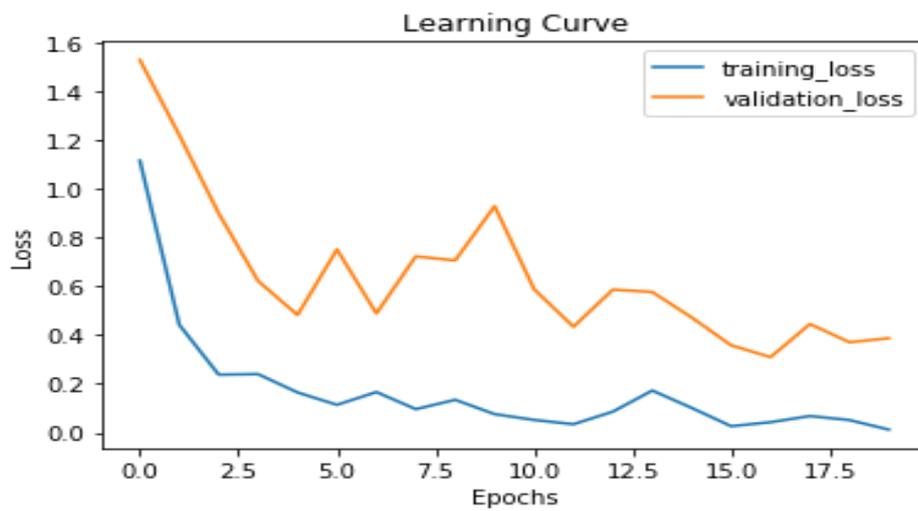
33/33 [=====] - 185s 6s/step - loss: 0.0512 -
accuracy: 0.9848 - val_loss: 0.3701 - val_accuracy: 0.7778

Epoch 20/20

33/33 [=====] - 189s 6s/step - loss: 0.0123 -
accuracy: 1.0000 - val_loss: 0.3865 - val_accuracy: 0.8889

8. 繪出訓練結果

繪出訓練集和驗證集兩條學習曲線（learning curve），也就是以時間為橫軸，以損失為縱軸，討論其兩者關係。如下圖為錯誤率及正確率的學習曲線，可以看出錯誤率及驗證錯誤率逐步下降，而正確率及驗證正確率逐步上升，表示沒有發生過度擬合（Overfitting）的問題。

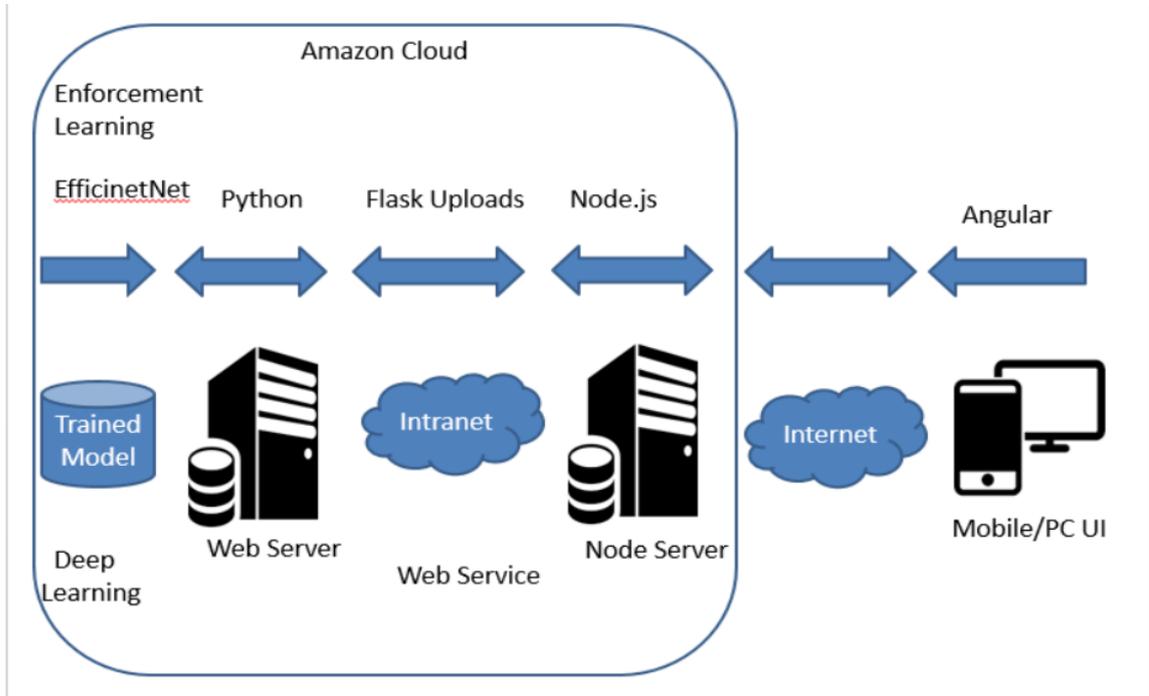


9. 用驗證集驗證與評估成效

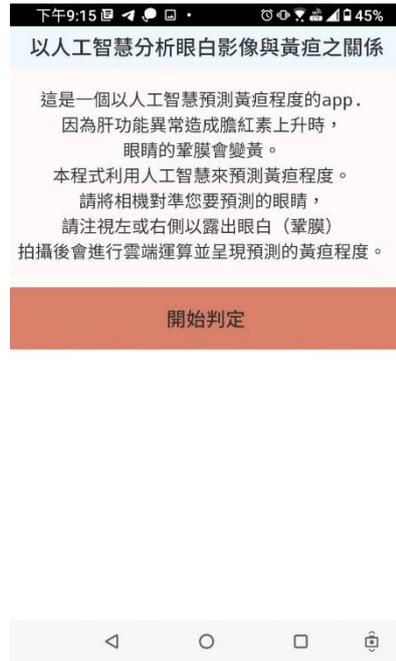
觀察驗證集的效果集進行分析，觀看準確率、實際值與預測值的分布、衡量不平衡。

(四) 軟體應用開發與設計

我們以 JavaScript 設計了一手機軟體，能夠實際以人工智慧分析眼白影像，初步依照訓練的資料庫診斷使用者是否有黃疸以及黃疸的嚴重程度。



前端為手機等使用者介面（UI），在此我們使用手機作為實際操作工具。使用者使用手機後鏡頭分別照取左右眼眼白，確認動作完畢後，點選上傳按鈕。拍攝圖像後，經過資料處理成易操作的資料格式後，藉用以連接前端及後端的 Angular 和 Node.js 平台，透過網際網路（Internet）傳送至 Node 伺服器（Node Server）。再透過內部網路（Intranet），以 Web 服務（Web Service）標準化後，傳送至網頁伺服器（Web Server），呼叫深度學習（Deep Learning）的訓練模型（Trained Model），經過算後，回傳結果至使用者介面。結果顯示使用者是否有黃疸以及預估的膽紅素區間（三個區間：0~1.2、1.3~3.5、3.6 以上），如附圖操作介面所示。



伍、研究結果

一、多元迴歸分析

若將預測結果及原始資料分成三群，預測成功的有 1966 筆資料點，成功預測率只有約 45.5% ($1966/4320*100\%$)。若將預測結果及原始資料分成兩群（只看膽紅素濃度是否過高），預測成功的有 2496 筆資料點，成功預測率約 57.78% ($2496/4320*100\%$)。由結果顯示使用多元迴歸分析時，將預測值及實際值分成兩群的成功預測準確度較高。

二、深度學習

將眼睛的影像經過資料篩選及影像前處理，再利用影像增強的技巧處理資料不平衡的問題以及利用已經訓練好的深度學習模型經由轉移學習調整參數，最後進行深度學習的訓練，正確率可高達 99% 及驗證正確率也有 89%，大幅提高此方法的可用性。

陸、討論

由以上研究方法可得知，最初我們會希望以科技取代人力判斷黃疸症狀，是為了能更客觀且更快速的獲得正確答案。以往醫生只能藉由經驗來猜測，而且還得經過儀器檢測才能得知數據。藉由多元迴歸分析或類神經網路所構成的程式碼，在未來或許可以開發成軟體，在檢測黃疸的同時還能將數據上傳，使資料庫不斷更新、並提供更為準確的結果。

一、RGB 與結果之關係

在分析圖片中眼白部分的 RGB 比例時，可由資料看出 R 和 B 與結果呈負相關，G 則為正相關。在公式中 R 係數為 -0.0090453 ，係數取絕對值數值相對 G 及 B 小很多，故 R 的權重比例非常少，而 G 和 B 取絕對值後的數值較大，皆達 0.06 以上，相對造成的影響較大。

二、不平衡資料

在此研究中，我們也遇到了許多醫療影像研究遭遇的共同難題，有疾病的影像遠少於健康者的影像，即為所謂的「不平衡資料」 imbalanced data。當我們要測驗機器是否準確的判斷病患是否有黃疸症狀時，通常是藉由判斷準確率（accuracy）來下結論。不過這也衍生了一個問題：所有病患中真正有黃疸症狀的僅是少數人，因此程式資料庫中正常人的眼白數據會占大多數。因此當一機器要預測輸入的數值時，猜測此數值為「非黃疸病患」的答對率會比「黃疸病患」的數據較為高，因而造成誤判實際準確度，這種現象稱為過度擬合（overfitting），發生在資料量較少的情況。

此問題常有兩種改善方法：設定權重及資料增強，選用其中一種即可。在此研究中使用資料增強。

（一）設定權重

計算原資料類別之反比，給予資料量少的類別高權重；給予資料量多的類別相對低的權重。

（二）資料增強

資料增強是增加資料量的方法，將資料圖片翻轉、縮放、改變色溫、調整亮度等等，用同一張圖片複製、調整成多張照片。面對 imbalanced data，對各類別取相對比例，並依其比例去增加其對應資料的資料量。使得資料量均勻、相等是理想狀態，但實際上當然無法達成，故實際狀態越接近均勻越佳。在本研究中我們以複製及選轉、平移圖像做為增加的方式。

三、轉移學習

訓練一個新的深度學習模型需要時間，可能需要幾天到幾個月，為了節省時間及人力，常使用的方法為轉移學習一用已建構好的架構訓練目標資料，其中可進行調整修飾，使其達到研究者的需求。以此研究中使用的 EfficientNet 為例，我們所用的 EfficientNetB5 版本在同領域中，算是計算速度非常快、參數量少、精準度極高、濾鏡可達約 2000 萬個，是非常優秀且為第一個超越人類辨識圖片的 AI 模型。故我們基於此模型上進行參數調適後，應用於本研究的深度學習上。

柒、結論

透過此研究，我們可以看出對於眼白黃的程度和血液膽紅素濃度的關聯性進行多元迴歸分析，得出的預測成功率只達 45.5% ~ 57.78%。若使用 CNN 進行監督式深度學習，預測結果準確率可達接近 88.89% ~ 99%。

我們希望未來能精進我們所設計的手機等行動裝置軟體，協助醫生篩檢病患黃疸症狀。此工具亦可貢獻於偏鄉醫療，促使民眾能及時就醫，且不需抽血即可預測高準確率的膽紅素濃度。

捌、參考資料及其他

壹、中文部分

【一本書】

鴻海教育基金會 (2019) · 人工智慧導論 · 新北市：全華圖書。

貳、網路資源

一、中文部分

【期刊文章】

高醫醫訓 (2015) · 黃疸 · *高醫醫訊月刊*, 35 · 取自

<http://www.kmuh.org.tw/www/kmcj/data/10410/9.htm>

台灣家庭醫學醫學會。取自 <https://www.tafm.org.tw/ehc-tafm/viewDocument>

【摘要及資料庫資料】

A+醫學百科 (2014 年 9 月 5 日) · 膽紅素 · 取自 [http://cht.a-](http://cht.a-hospital.com/w/%E8%83%86%E7%BA%A2%E7%B4%A0)

[hospital.com/w/%E8%83%86%E7%BA%A2%E7%B4%A0](http://cht.a-hospital.com/w/%E8%83%86%E7%BA%A2%E7%B4%A0)

高點醫護網 · 取自 <http://doctor.get.com.tw/m/Journal/detail.aspx?no=403551>

台灣人工智慧學校 · 取自 <https://aiacademy.tw/>

【單篇文章】

蔡聰明 (2017 年 9 月) · 臺北市：中央研究院 · 最小平方方法與迴歸分析 · 取自

<https://w3.math.sinica.edu.tw/mathmedia/HTMLarticle18.jsp?mID=41303>

Lynn (2017 年 7 月 19 日) · 監督式學習？增強學習？聽不懂的話，一定要看這篇入門的機器學習名詞解釋 · 取自 <https://www.inside.com.tw/article/9945-machine-learning>

曲建仲 (2017 年 10 月 5 日) · 翻轉人類未來的 AI 科技：機器學習與深度學習 · 取自

<https://technews.tw/2017/10/05/ai-machine-learning-and-deep-learning/>

ianlini (2017 年 3 月 14 日) · 我處理 Imbalanced Data 的心得 (上) · 取自

<https://ianliniblog.wordpress.com/2017/03/14/%E6%88%91%E8%99%95%E7%90%86-imbalanced-data-%E7%9A%84%E5%BF%83%E5%BE%97-%E4%B8%8A/>

Jason Kuan (2017 年 8 月 26 日) · 【講講科普】 當你有了三個孩子他們分別叫監督式學習、非監督式學習與強化式學習 · 取自 <https://medium.com/@capillaryj/機器學習-當你有了三個孩子他們分別叫監督式學習-非監督式學習與強化式學習-1caa26c7da87>

Tommy Huang (2018 年 3 月 27 日) · 卷積神經網路(Convolutional neural network, CNN) — CNN 運算流程 · 取自 <https://medium.com/@chih.sheng.huang821/%E5%8D%B7%E7%A9%8D%E7%A5%9E%E7%B6%93%E7%B6%B2%E8%B7%AF-convolutional-neural-network-cnn-cnn%E9%81%8B%E7%AE%97%E6%B5%81%E7%A8%8B-ecaec240a631>

二、英文部分

【摘要及資料庫資料】

GitHub. EfficientNet. From <https://github.com/qubvel/efficientnet>

TensorFlow. From <https://www.tensorflow.org/>

【單篇文章】

Zichen Wang. Practical tips for class imbalance in binary classification. Medium. Retrieved August 10, 2018, from

<https://towardsdatascience.com/practical-tips-fo-class-imbalance-in-binary-classification-6ee29bcdb8a7>

Umer Farooq. Web Services in Python. Medium. Retrieved February 26, 2018, from [Web Services in Python - Umer Farooq](#)

Arseniy Tomkevich. Simple Application with Angular 6 + Node.js & Express. Medium. Retrieved December 25, 2018, from [Simple Application with Angular 6 + Node.js & Express](#)

【評語】 052503

此作品使用多元迴歸分析和 CNN 深度學習兩種技術來建立眼白黃的程度和血液膽紅素濃度的關聯性。目標是能根據一張眼白黃的相片來正確預測對應的血液膽紅素濃度。所用來訓練模型的相片都需要先經過一些前處理。此作品與中國醫藥大學合作，在資料收集上很用心。唯其套用現成 efficientNetB5 model，在機器學習方法上的創新及探討程度較為不足。

動機與目的

黃疸為許多代謝相關疾病的症狀，最典型的體徵為皮膚及眼白變黃。由於部分人原膚色就偏黃或誤認變黃為曬黑等而疏忽了疾病，當診斷出黃疸或器官病變時病情已加劇。加上偏鄉地區醫療資源分配不均日益加重，應尋改善方法。另外，人工智慧建立在電腦計算，而電腦現今的運算速度及效率已超越人類。以行動裝置運用人工智慧分析眼白影像與黃疸之關係能夠協助醫療機構、診所做篩檢，大幅減少程序與人力，提升效率。

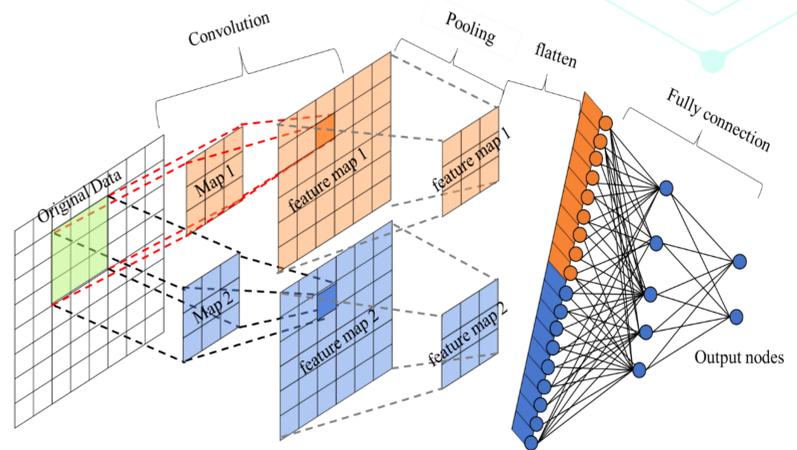
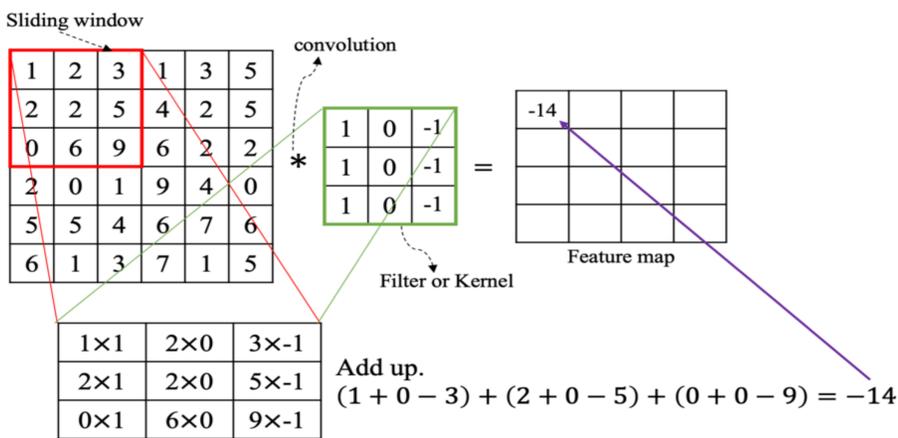


黃疸的成因

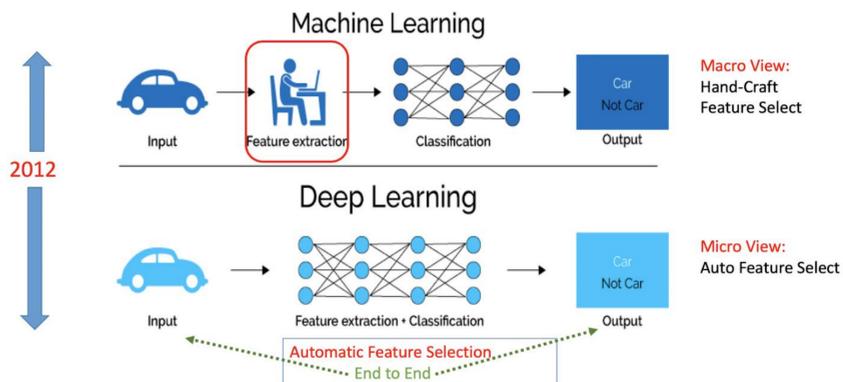
黃疸可能源自肝功能受損或溶血疾病。體內的紅血球自然代謝後，細胞內的血色素就會產生膽紅素。其中會先產生「未結合型膽紅素 (Unconjugated bilirubin)」，又稱「間接型膽紅素 (Indirect Bilirubin)」，接著隨血液流進肝臟代謝，並在肝細胞中與葡萄糖醛酸結合，形成「結合型膽紅素 (Conjugated bilirubin)」，或稱為「直接型膽紅素 (Direct Bilirubin)」。結合型膽紅素最後會由糞便及尿液排出體外。因此若是膽紅素沒有代謝完全就會積在體內造成體色泛黃。

人工智慧

- **人工神經網路 (Artificial Neural Networks, ANN)** 又稱為類神經網路，啟發於生物神經網路的運作方式。類神經網路運用大量人工神經元建構機器學習 (Machine Learning) 的計算模型，建立神經元間的連結，而透過計算的過程及結果，對模型本身做調整，達成學習的效果。
- **類神經網路** 分為三個階層：輸入層 (Input layer)、隱藏層 (Hidden layer)、輸出層 (Output layer)。輸入層為眾多接收刺激訊息的神經元 (neuron)，將接收到的訊息傳至下一層。隱藏層可為單層或多層，夾在輸入層及輸出層之間。隱藏層及各隱藏層中的神經元數目是影響神經網路學習的重要因素。
- **機器學習** 的模型建立在訓練範例上。分為監督式學習 (supervised learning) 和非監督式學習 (unsupervised learning)。監督式學習必須事先給定標籤 (label，分類的結果)，由訓練資料中得出規則，進而預測新的目標對象；非監督式學習不需要事先給定標籤，能自動對輸入資料進行分類分群，得出規律。此研究使用監督式學習。
- **卷積神經網路** 包括卷積層 (convolution layer) 及池化層 (pooling layer)。辨識照片時會設定過濾器 (filter) 擷取照片中某特徵，不同過濾器能識別不同目標特徵。在卷積層中，過濾器掃描圖片，紀錄圖片中不同區塊的特徵強度。在池化層中，將特徵圖 (feature map) 分為多個區域，再選出特徵強度最強的數值 (最大化、平均化、最小化等) 代表此區域，將特徵圖降維。卷積神經網路可做多次卷積和池化，最後接上全連接層 (fully connected layer) 後輸出。



- **預處理程序 (preprocessing)** 能使電腦更有效學習。空間濾波器 (spatial filtering) 是預處理程序的一種技巧，以卷積運算凸顯某特徵，使用不同濾波器 (filter) 有不同效果。
- **深度學習 (deep learning)** 是表徵學習的演算法，特質為能自行選取特徵取代人工選取。因 GPU 能快速運算，不再需要人工給予標籤，神經網路能自行擷取特徵值將其分類、預測結果，達到「端到端 (end to end)」的學習。
- **訓練深度學習模型時**，將資料庫分為三群：訓練集 (training)、驗證集 (validation)、測試集 (testing)。模型運用訓練集進行學習，針對圖片調整參數，使其能辨識這些圖片。驗證集提供模型驗證，調適自身結構，如神經元個數。最後模型利用測試集檢驗辨識分類效果。



蒐集資料

我們和中國醫藥大學附設醫院合作，經病人同意後，分別拍攝病人的左右眼，取得總計354張照片，並收集病人的實際膽紅素濃度資料。

每位病患皆會拿著一張標準色卡以確認相片最亮點較為一致。標準色卡的白色為陶瓷所燒製而成，較不會隨著時間改變而導致顏色有誤差。



人工智慧分析

1. 匯入相關模組：

匯入os (檔案建置與開啟)、numpy (矩陣計算)、pandas (整理資料表格與前處理)、matplotlib.pyplot (繪製圖形)、opencv (影像前處理)等模組。

2. 資料匯入與處理：

從354張照片篩選解析度佳者，並框選出眼睛部分，使其為較易後續處理的形式。最後選出113張照片作為本次深度學習的樣本。

3. 歸類照片類別：

將照片歸類成三個類別：膽紅素濃度0~1.2 (51張)、1.3~3.5 (30張)、3.6以上 (9張)，並控制圖片大小、宣告照片的標題、將指定的影像放入呼叫的畫圖位置等。

4. 影像前處理 (preprocessing)：

(1) 鏡射 (Reflection)

a. 裁切 (Cropping)：將每張照片調整成相同大小。

b. 鏡射補值：以長邊為直徑，短邊以鏡射方式補值，直到長寬等長。

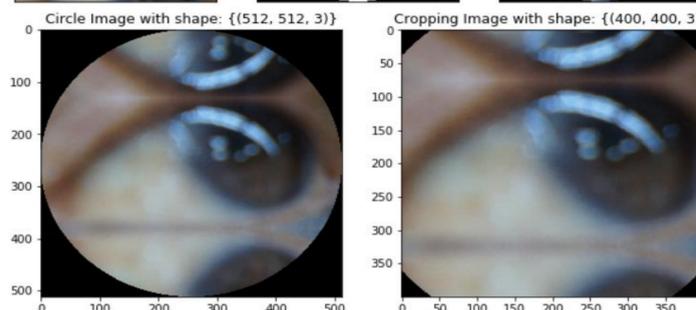
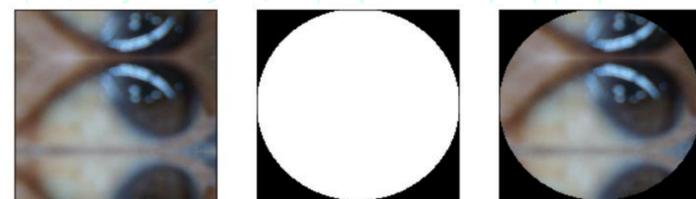
c. 將照片裁成圓形：以邊為直徑做一相切圓 (正方形的內切圓)，切除圓以外的影像，只保留圓內的影像做處理。裁剪完後確保每一張照片為同一大小，將每張照片設為一特定大小。

(2) 增加影像面積，減少不必要的特徵：將調整為同一大小的影像再次做裁切。以中心點取一特定值的正方形，只保留正方形內的影像，減少不必要的特徵。

(3) Image Processing 影像處理：一般在CNN中，用多種濾波器擷取各種特徵值做運算，類似傳統影像處理人工選擇濾波器，例如範例。

在此為深度學習，不需人工選擇濾鏡，而是CNN自行產生最符合的濾鏡

類別	數值	影像圖片	
1	0~1.2		
2	1.3~3.5		
3	3.6~		



種類	原始	直方圖Histogram	直方圖均勻化 Clahe (Adaptive Histogram)	高斯 Gaussian
圖片				

5. 增強預測成效：

(1) 資料增強 (data augmentation)：將影像水平平移、垂直平移、0°~180°任一角度旋轉、任意水平翻轉、任意垂直翻轉、空隙補常數、設定常數值為0，達到增加資料量的效果。

(2) 轉移學習 (transfer learning)：套用已訓練好的模型，修改部分使其符合此研究。在此我們使用EfficientNetB5進行轉移學習。

6.開始訓練：共進行二十個回合的訓練，每個回合總共重複33次的調整權重及參數，底下列出每一回合的正確率 (accuracy)、錯誤率 (loss)、驗證正確率 (validation_accuracy) 及驗證錯誤率 (validation_loss)，正確率可高達99%，驗證正確率也有89%。

Epoch 1/20 33/33 386s 12s/step - loss: 1.1176 - accuracy: 0.5954 - val_loss: 1.5267 - val_accuracy: 0.3333	Epoch 19/20 33/33 185s 6s/step - loss: 0.0512 - accuracy: 0.9848 - val_loss: 0.3701 - val_accuracy: 0.7778
Epoch 2/20 33/33 347s 11s/step - loss: 0.4390 - accuracy: 0.8550 - val_loss: 1.2188 - val_accuracy: 0.4444	Epoch 20/20 33/33 189s 6s/step - loss: 0.0123 - accuracy: 1.0000 - val_loss: 0.3865 - val_accuracy: 0.8889

多元迴歸分析

多元線性迴歸分析 (Multiple Regression Analysis) 為探各討變數之關係的迴歸模型，一般模型為：

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p + e$$

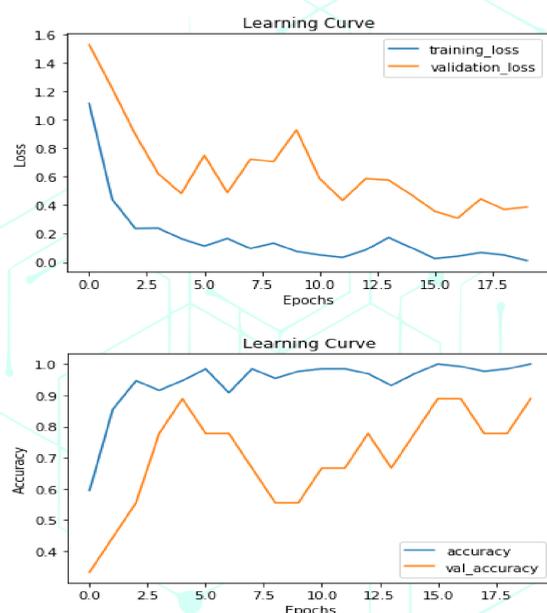
在此研究程式中，我們將R、G、B三數值設為X變數，並將膽紅素值設為Y變數。 β_0 為Y截距， $\beta_1 \sim \beta_p$ 則為未知且須估計的迴歸係數。我們以最常見的分析方法：最小平方法 (Least squares) 求得 β 值，並運用Excel多元迴歸分析的預設工具達到此目的。 e 為數值與起始值間的差距。所以在此研究中，我們將尋求多元迴歸分析的目標函數或是損失函數 (loss function)，也就是找到一組最佳的參數，使得函數值最小。

	係數	標準誤	t 統計	P-值	下限 95%	上限 95%	下限 95.0%	上限 95.0%	
16									
17	截距	1.79009078	0.10775769	16.6121858	4.0047E-60	1.57883046	2.00135109	1.57883046	2.00135109
18	R	-0.00904534	0.00193041	-4.68571426	2.8756E-06	-0.01282993	-0.00526075	-0.01282993	-0.00526075
19	G	0.06795248	0.00376582	18.044533	3.0956E-70	0.06056954	0.07533542	0.06056954	0.07533542
20	B	-0.06244712	0.00236849	-26.3657526	3.406E-142	-0.06709059	-0.05780366	-0.06709059	-0.05780366

依係數可得出公式： $Y = 1.79009078 + (-0.0090453) R + (0.06795248) G + (-0.0624471) B$

研究結果

繪出訓練結果：繪出訓練集和驗證集兩條學習曲線 (learning curve)。以時間為橫軸，以損失為縱軸，討論兩者關係。從錯誤率及正確率的學習曲線，可看出錯誤率及驗證錯誤率逐步下降，正確率及驗證正確率逐步上升，表示沒發生過度擬合Overfitting。

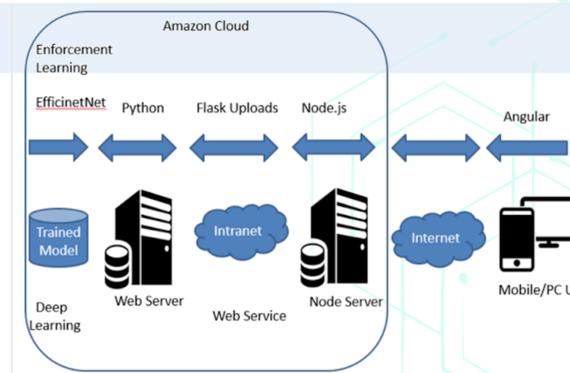


1. 多元迴歸分析：將預測結果及原始資料分成三群，預測成功的有 1966 筆資料點，成功預測率僅45.5%。將預測結果及原始資料分成兩群，預測成功的有 2496 筆資料點，成功預測率 57.78%。結果顯示使用多元迴歸分析時，將預測值及實際值分成兩群的成功預測準確度較高。

2. 深度學習：將眼睛影像經資料篩選及影像前處理，再利用影像增強技巧處理資料不平衡問題，並利用已訓練好的深度學習模型經由轉移學習調整參數，正確率高達99%及驗證正確率89%。

軟體應用開發與設計

我們以JavaScript設計了一手機軟體，能夠實際以人工智慧分析眼白影像，初步依照訓練的資料庫檢斷使用者是否有黃疸以及黃疸的嚴重程度。結果顯示使用者是否有黃疸以及預估的膽紅素區間 (三個區間：0~1.2、1.3~3.5、3.6以上)。



討論與結論

一、RGB與結果之關係：分析圖片中眼白RGB比例時，可看出R係數絕對值相對G及B小很多，故R的權重非常少，而G和B絕對值較大，皆達0.06以上，相對影響較大。

二、不平衡資料：於研究中遇到了許多醫療影像研究遭遇的難題：有疾病的影像遠少於健康者的影像，即所謂的「不平衡資料」。不過這衍生了一個問題：所有病患中真正有黃疸的僅少數，因此資料庫中正常人的眼白數據占多數。當機器預測輸入的數值時，猜測數值為「非黃疸病患」的答對率會比「黃疸病患」的數據高，造成誤判。這種現象稱為過度擬合 (overfitting)，發生在資料量較少的情況。此問題常有兩種改善方法：設定權重及資料增強。此研究中使用資料增強。

(一) **設定權重：**給予資料量少的類別高權重；給予資料量多的類別低權重。

(二) **資料增強：**增加資料量，將圖片翻轉、縮放、改變色溫、亮度，用同張圖片複製調整成多張照片。實際資料量越均勻越接近理想狀態。此研究中以複製、旋轉、平移圖像增加資料。

三、轉移學習：訓練新的深度學習模型耗時，為節省時間及人力，常使用轉移學習：用已建構好的架構訓練目標資料，可調整修飾使其符合研究者需求。研究中使用的 EfficientNetB5是極優秀且為第一個超越人類辨識圖片的 AI 模型。

四、結論：從研究看出對於眼白黃的程度和血液膽紅素濃度的關聯性進行多元迴歸分析，得出的預測成功率僅45.5% (三群) ~ 57.78% (二群)。使用 CNN 進行監督式深度學習，預測結果準確率達99%，驗證準確率達88.89%。