

南華大學科技學院資訊管理學系

碩士論文

Department of Information Management

College of Science and Technology


Nanhua University

Master Thesis

運用直交表優化卷積神經網路－以文字辨識為例

Optimizing Convolutional Neural Network by Orthogonal Array--

A Case Study of Text Recognition



張庭彰

Ting-Chang Chang

指導教授：陸海文 博士

Advisor: Hai-Wen Lu, Ph.D.

中華民國 111 年 6 月

June 2022

南華大學
科技學院資訊管理學系
碩士學位論文

運用直交表優化卷積神經網路-以文字辨識為例
Optimizing convolutional neural network by orthogonal array
-A Case Study of Text recognition

研究生：張庭軒

經考試合格特此證明

口試委員：
劉信益
毛怡文
陸海文

指導教授：陸海文

系主任(所長)：陳信良

口試日期：中華民國 111 年 6 月 11 日

南華大學資訊管理學系碩士論文著作財產權同意書

立書人：_____張庭彰_____之碩士畢業論文

中文題目：
運用直交表優化卷積神經網路-以文字辨識為例

英文題目：

Optimizing the Convolutional Neural Network for Text
Recognition Using the Orthogonal Array

指導教授： 陸海文 博士

學生與指導老師就本篇論文內容及資料其著作財產權歸屬如下：

- 共同享有著作權
- 共同享有著作權，學生願「拋棄」著作財產權
- 學生獨自享有著作財產權

學 生：_____張庭彰_____ (請親自簽名)

指導老師：_____陸海文_____ (請親自簽名)

中華民國 111 年 6 月 15 日

南華大學碩士班研究生

論文指導教授推薦函

資訊管理學系碩士班 張庭彰 君所提之

論文

運用直交表優化卷積神經網路-以文字辨識為例

係由本人指導撰述，同意提付審查。

指導教授

陸海文

111年6月15日

運用直交表優化卷積神經網路—以文字辨識為例

學生：張庭彰

指導教授：陸海文 博士

南 華 大 學 資 訊 管 理 學 系 碩 士 班

摘 要

影像辨識一直以來都是機器學習的主流之一，傳統機器辨識與訓練，多採取人工特徵擷取法，然而運用此方法要找出強健的特徵去使模型學習處理數十百萬的資料是不容易的，隨著深度學習、卷積神經網路的開發，文字辨識已運用在日常生活各領域，但科學家對此機器的辨識速度與精準度的要求也不斷提高。

本研究採用卷積神經網路程式為研究範例，透過田口品質工程的理念，尋找影響其品質特性之控制因子與變動水準數，製作 L18 直交表制定實驗順序，搭配 Google Colaboratory 軟體執行取得數據，依據實驗結果計算其 S/N 比，繪製出最佳化控制因子反應圖，其中以輸出層激勵函數採用 Softmax，池化層為 Pool_size(2, 2)，卷積層為 Tanh，拋棄層為 0.8，隱藏層激勵函數採用 Elu，隱藏層神經元為 256，訓練方式優化器為 Adam，訓練方式準確率模型為 Binary_Accuracy 為最佳組合，最後進行確認實驗驗證結果，以最佳化組合平均值為

0.9971，S/N 比為-0.025，優於其他組合結果，確認該組合為最佳組合，為以供爾後研究人員做為參考。

關鍵詞：機器學習、卷積神經網路、田口品質工程



Optimizing Convolutional Neural Network by Orthogonal Array-- A Case Study of Text Recognition

Student: Ting-Chang Chang

Advisor: Hai-Wen Lu, Ph.D.

Master Program: Department of Information Management, College of Science
and Technology, Nanhua University

Abstract

Image recognition has been a mainstream topic in machine learning. Conventionally, machine recognition and training has been conducted through manual feature extraction, which requires completing the challenging task of finding robust features for models to learn to read millions of data. Following the introduction of deep learning and convolutional neural network, text recognition technology has been integrated in daily living. However, the scientific demand for the speed and accuracy of machine recognition continues to increase. With technology evolving continually, mass data integration and application poses a major challenge in machine learning, in which the selection of parameters in algorithms is critical.

This study will focus on the convolutional neural network program. Through Taguchi quality engineering, the control factors to and levels of changes in the quality of a convolutional neural network will be explored. An L18 orthogonal array will be established to formulate the experiment procedure. Data will be collected using Google Colaboratory. Signal-to-noise ratio and quality characteristics will be calculated according to the experiment results, and the effects of various factors will be explained. The experiment will be repeated with the parameters of the control factors adjusted to optimize the factor set.

Keywords: machine learning, convolutional neural network, Taguchi quality engineering, Google Colaboratory

目 錄

碩士論文著作財產權同意書.....	I
指導教授推薦函	II
摘要.....	III
Abstract	V
目錄	VII
圖目錄.....	IX
表目錄	X
第一章 緒論	1
1.1 研究背景與動機	1
1.2 研究目的	1
1.3 研究架構與流程	2
第二章 相關文獻	4
2.1 機器學習	4
2.2 深度學習	5
2.3 卷積神經網路	6
2.4 激勵函數.....	10
2.5 優化器(Optimizer).....	12
2.6 準確率(Accuracy).....	12
2.7 田口品質工程	13
2.8 品質損失函數	13
2.9 訊號雜音比 (Signal-to-Noise , S/N 比)	15

第三章 研究方法	17
3.1 決定實驗品質特性.....	18
3.2 決定影響品質特性之控制因子與其變動水準數.....	18
3.3 計算自由度並套用適合的直交表.....	21
3.4 執行實驗並收集數據	22
3.5 依據實驗結果計算 S/N 比、平均值與繪製其反映圖，找出最佳化控制因子 組合	25
第四章 實驗結果分析.....	29
第五章 結論.....	32
參考文獻.....	34

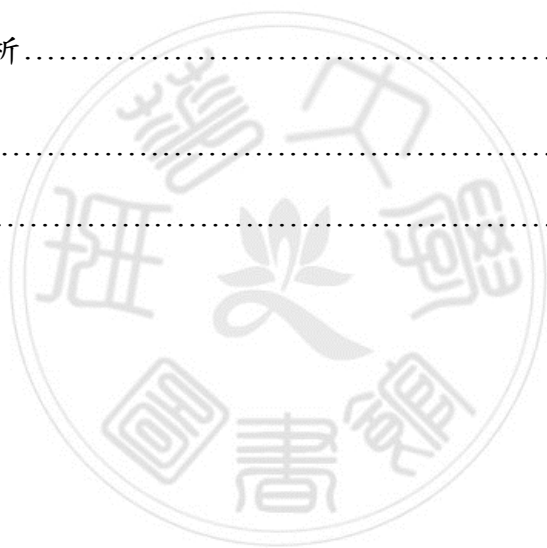


圖 目 錄

圖 1-1	研究架構圖.....	3
圖 2-1	深度神經網路.....	6
圖 2-2	卷積神經網路釋義圖.....	7
圖 2-3	卷積運算釋義圖.....	8
圖 2-4	池化運算釋義圖.....	9
圖 2-5	全連接層運算釋義圖.....	10
圖 2-6	品質特性表.....	15
圖 3-1	實驗架構圖.....	17
圖 3-2	執行實驗畫面(1).....	23
圖 3-3	執行實驗畫面(2).....	24
圖 3-4	執行實驗畫面(3).....	24
圖 3-5	最佳化因子 S/N 反應圖.....	27
圖 3-6	最佳化因子平均值反應圖.....	28

表 目 錄

表 2-1	品質損失代號與含意.....	14
表 2-2	訊號雜音比公式.....	16
表 3-1	實驗品質控制因子與其水準參數.....	20
表 3-2	L18 田口直交表表.....	22
表 3-3	卷積神經網路實驗準確率紀錄表.....	25
表 3-4	S/N 比與平均值計算結果.....	26
表 3-5	最佳化因子 S/N 反應表.....	26
表 3-6	最佳化因子平均值反應表.....	27
表 4-1	最佳化因子實驗數據.....	31

第一章 緒論

1.1 研究背景與動機

自從網路發展與智慧產品的問世下，人們逐漸仰賴科技所帶來的便利性，也同時希望科技的進步可以使得機器互能貼近人性，所以人工智慧與機器學習將是未來重要的趨勢之一。目前在全球各個國家已將人工智慧與機器學習逐漸運用在生活的各個領域上，由於智慧行動裝置的普及，每分每秒將產生出大量的數位影像資料與數據，若能善用這些資料數據並透過機器學習，機器將可以沿生出另一種更貼近生活服務的模式，使得生活更加便利。目前機器手臂、無人機自動駕駛就是機器學習的成果[1]。

在機器學習的領域中，影像辨識是相當廣泛被運用的。由於機器學習需要大量的資料，而且這些資料必須先行註記，才能訓練機器透過演算法得到正確結果，如此將大大耗費時間與人力。本研究希望透過田口式品質工程的理念，找出較能廣泛使用的演算法，提升辨識度的同時，也可降低生產成本[2]。

1.1 研究目的

延續研究背景與動機內容提到，影像辨識已廣泛被運用生活中，

其中文字辨識是影像辨識領域裡的其中一部份。本論文研究主題為如何以田口品質工程之方法結合 Google Colaboratory 之軟體並採用 python 機器學習與深度學習特訓班此教材內容程式，找出對於文字辨識此領域另有其最佳化參數條件。本研究將探討在文字辨識中影響品質特徵值的控制因子與水準，繪製成田口直交表，利用 Google Colaboratory 軟體去取得最佳化組合數據，爾後將其計算出 S/N 比、反應圖表並選擇最佳化的組合。本研究目的希望透過實驗的成果，可作為爾後研究人員之參考數據之一。

1.2 研究架構與流程

依照研究背景動機與目的，制定研究架構。透過相關文獻的探討研究，找出本實驗的品質特性與步驟並進行實驗，再依據實驗數據繪製圖表進行結果分析，最後以分析後的結果作本次實驗的結論。研究架構圖如圖 1.1 所示。

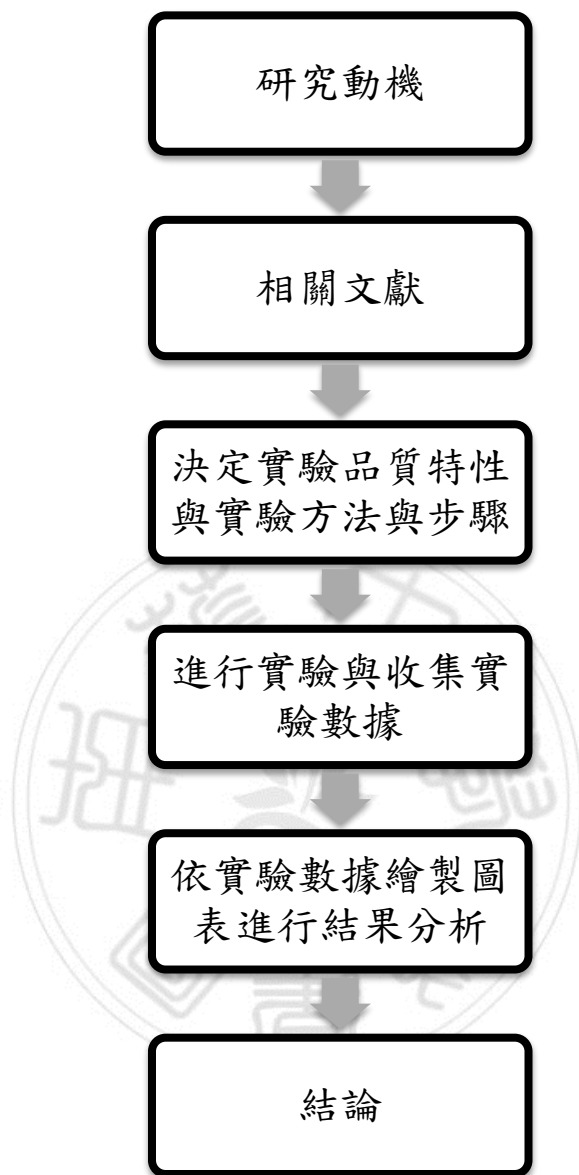


圖 1.1 研究架構圖

第二章 相關文獻

2.1 機器學習

機器學習(ML)為人工智慧(AI)的一部分，其中是以訓練電腦由資料庫中反覆運算整理，再根據經驗改善，透過各種演算法來歸納資料庫的關聯性與模式，進而發展出可精準分析的資料模型；此模型經由存取的資料與經驗越多，機器學習預測的結果也會越準確。目前機器學習廣泛運用於日常當中，未來將會成為生活所不可分離的一部分。

機器學習擁有不同的學習模式，依照資料的本質與要求之成果，分別採取監督式、半監督式、非監督式、強化式四種學習模式[2] [3]。

(一)監督式學習：

使用資料集將先行註記再輸入機器，讓機器透過演算法學習如何辨識資料間的差異與相似度，而判斷出準確的結果。

(二)非監督式學習：

輸入機器前的資料大多為未註記，機器只能利用資料間關聯性去進行資料分類整理，其判斷結果與預期結果可能會有所落差。

(三)半監督式學習：

介於監督式學習與非監督式學習方式之間，輸入機器前將輸入的

資料少量先行註記，再讓系統進行運作，此學習模式可使機器提升學習速度，使其判斷的結果更加準確。

(四)強化式學習：

輸入機器前的資料與非監督式學習同樣多為未註記，機器採取結果的好壞不斷反饋，透過演算法逐步修正，將取得最佳利益的結果。

2.2 深度學習

深度學習為機器學習其中的分支，透過模仿人腦神經網路的運作方式，使用多層次的方式來擷取資料的特徵，獲取有用的訊息並解決問題。相比機器學習之下，深度學習無須人工資料註記與分類，以建構類神經網路的方式執行部分學習後，人工智慧就可自行進行高度的機器學習。目前深度學習已應用於視覺辨識、語音識別等領域，而最常見的架構分別為：深度神經網路 DNN(Deep Neural Network)、卷積神經網路 CNN(Convolutional Neural Network)、遞迴神經網路 RNN(Recurrent Neural Network)。其中以卷積神經網路 CNN 最常運用其影像辨識上[2]。

以圖 2-1 所示：

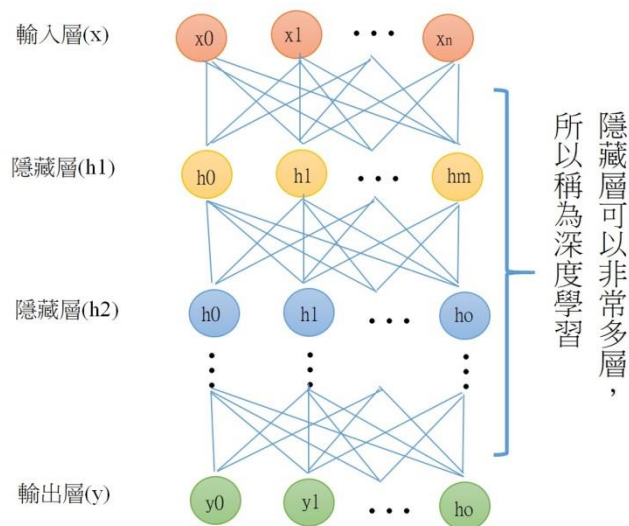


圖 2-1 深度神經網路

資料來源: <http://tensorflowkeras.blogspot.com/>

2.3 卷積神經網路

卷積神經網路(CNN)又被稱為 CNNs 或 ConvNets，是目前深度神經網路領域用來做影像處理發展的主力。在傳統神經網路辨識與訓練，多採取人工特徵擷取法，然而運用此方法要找出強健的特徵去使模型學習處理數十百萬的資料是不容易的，除此之外，資料通常包含水平、垂直、顏色等二維或三維資訊，而傳統神經網路輸入處理為一維資料，故在影像處理中將受到極大限制。卷積神經網路較傳統神經網路於架構上最大差異為前者多建置卷積層、池化層與連接層三層。此神經網路透過卷積與池化兩方式自主學習擷取其資料特徵，並以權重共享的特點，考量的參數比起其餘神經網路更少，訓練速度更快，最後再透過全連接層分類

出結果[4] [6]。以圖 2-2 所示：

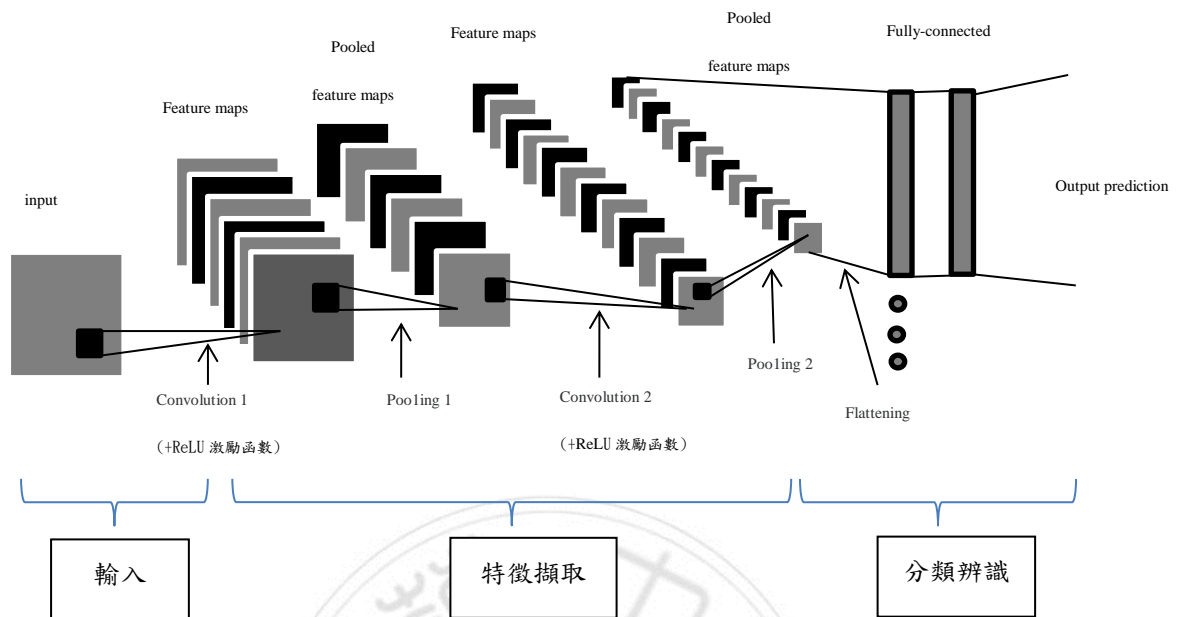


圖 2-2 卷積神經網路釋義圖

(一)卷積層

卷積層主要目的是為了保留輸入圖像的空間排列，並取得局部圖像作為輸入特徵，其原理以圖像的每一點為中心，透過不同的卷積核對點進行卷積運算，以由上而下、由左至右滑動的方式，計算其加權總和，加權總和結果會再加上激勵函數進行非線性轉換，最後得出特徵圖[2] [6]。以圖 2-3 所示：

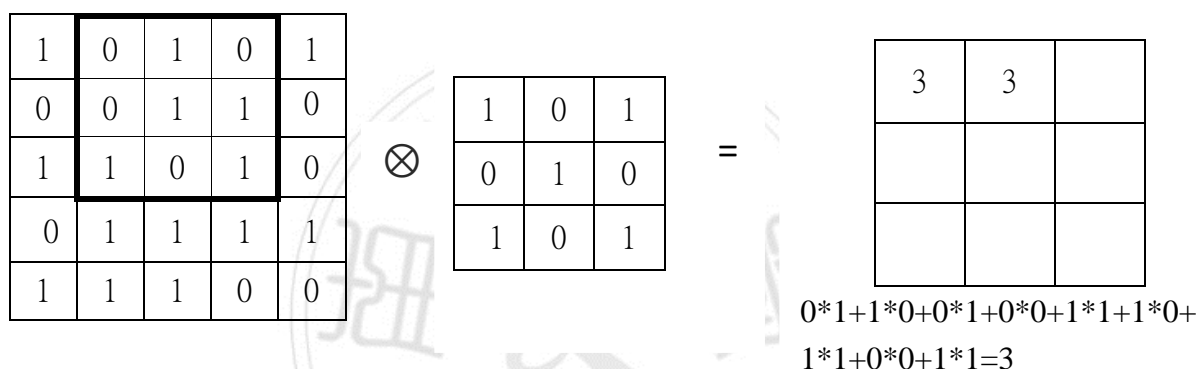
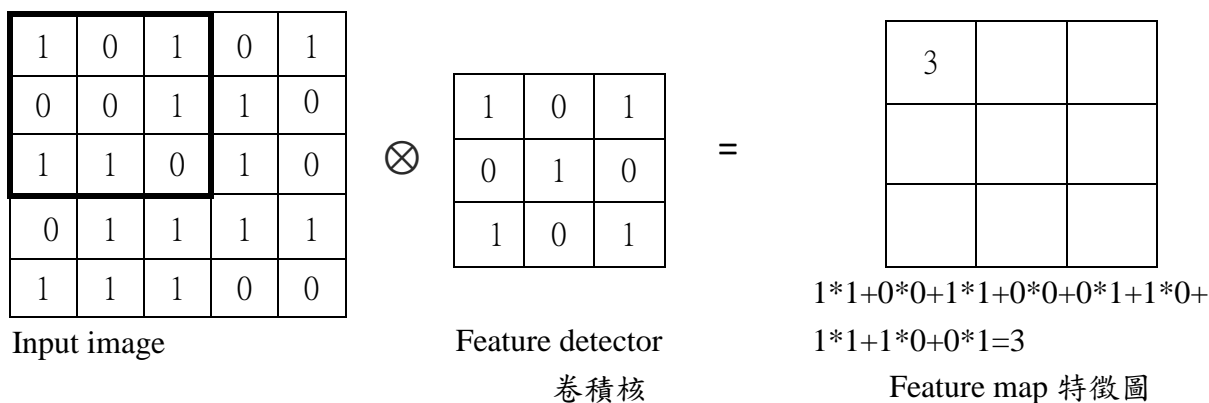


圖 2-3 卷積運算釋義圖

(二)池化層

池化層會將卷積層所產生的特徵圖加以保留其特徵，雖然會使原特徵圖縮小，但可提高神經網路之運算速度，擁有抗雜訊功能。一般池化層常用的方式有最大池化(Max Pooling)與平均池化(Average Pooling)，其滑動運算類似卷積層運算方式，兩者差異於池化層滑動方式並不會覆蓋前一覆蓋處，並取覆蓋處之最大值或平均值[2][6]。以圖 2-4 所示：

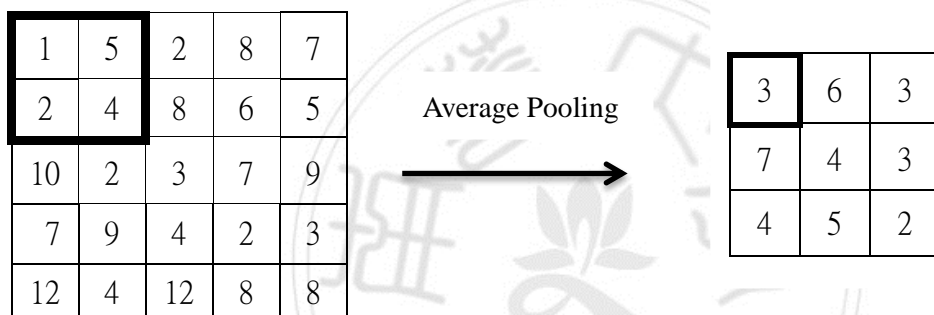
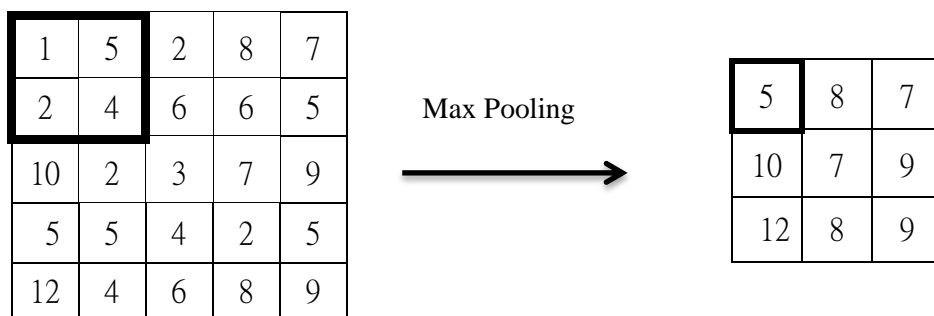


圖 2-4 池化運算釋義圖

(三)全連接層

輸入圖像經過卷積層與池化層的運算後，將原本三維特徵圖降維至一維的特徵向量進入全連接層，全連接層將透過隱藏層來進行特徵分類，通常使用 softmax 激勵函數來進行輸出預測，再經過反向傳播訓練得到最終判斷[2]。以圖 2-5 所示：

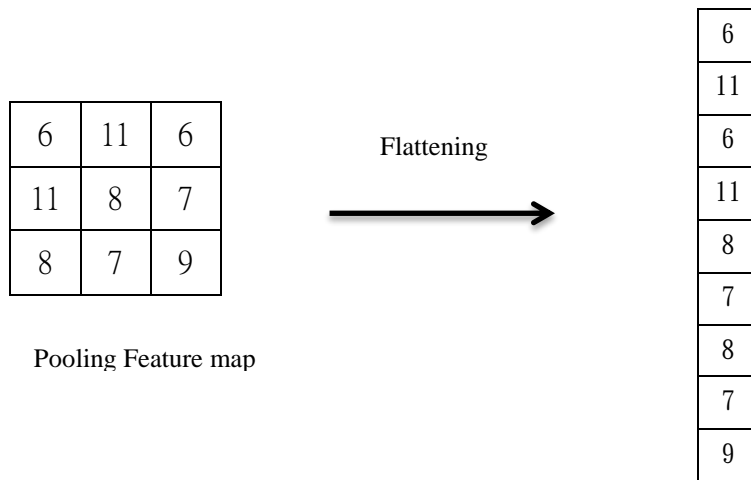


圖 2-5 全連接層運算釋義圖

2.4 激勵函數

類神經網路的發明構想原至於生物的神經網路，科學家為了使機器可模仿神經網路裡傳遞訊息的神經元，在運算模型中加入激勵函數，如此輸入值便會經過無數激勵函數的運算，得到最終的結果。最初類神經網路模型屬於二分法的感知器，感知器僅能處理線性問題，而事實上所有的問題大部分皆屬於非線性問題，因此逐漸發展出不同非線性激勵函數來改善神經網路的學習方法。本實驗主軸以不同激勵函數來作為控制因子的水準，以下將逐一介紹其激勵函數。

(一) Sigmoid

屬於初期神經網路運用最廣泛的激勵函數，其優點具有非線性可

微分的特性，任何變數經過 Sigmoid 接收輸出後皆可控制 $[0,1]$ 之間，方便訓練模型做二分法，缺點為其一是運算時間較耗時；其二為神經網路進行誤差反向傳遞時，需不斷以導數形式回傳來改變權重參數，隨著網路層數遽增，導數後將趨近於 0，訊息將無法進行反向傳遞，稱為梯度消失。其三輸出不是 zero-centered，在誤差反向傳遞時，權重的更新將影響模型收斂速度。

(二) Tanh

相比 Sigmoid 函數之下，Tanh 輸出屬於 zero-centered，故收斂速度較 Sigmoid 快，但還是有梯度消失與運算時間較慢的缺點。

(三) Softmax

對應 Sigmoid 二分類函數比較起來，Softmax 屬於多分類函數，而分類出的數值結果會以相對概率呈現，且所有輸出的概率和為 1，概率越大者，代表離目標值越高。

(四) Relu

其中文全名為線性整流函數，是近代神經網路最常使用的激勵函數，相比 Sigmoid 與 Tanh 函數，在計算方面上更加簡單，就以 Relu 導數來說，輸入值大於 0，輸出值為 1；輸入值小於 0，輸出值為 0，如此 Relu 函數可解決正區間梯度消失的缺點，故收斂速度較兩者快。

但相對來講，負區間的輸出值為 0 在進行反向傳遞時，也將會影響到部分對應參數的更新，進而造成神經元喪失功能(Dead relu)。

(五)Elu

為其改善 Relu 函數的缺點，Elu 函數計算加入可調整的指數項，使得在保有 Relu 函數的優點情況下，也使得負區間的數值可被控制並利用。

2.5 優化器(Optimizer)

神經網路進行誤差反向傳遞時，常以梯度下降法(導數)的方式回傳，藉此改變權重降低誤差，由於回傳參數的不同所因應的權重也不盡相同，所以優化器的選擇對於自適應學習率影響是相當大的。本研究選擇 Adam、Adadelta、Adagrad 三種不同優化器，藉此研究何種優化器較適用於本實驗。

2.6 準確率 (Accuracy)

機器學習中的準確率代表著運算模型的好壞，故本研究採用 Keras 開源神經網路資料庫中的 Accuracy、Binary_Accuracy、Categorical_Accuracy 三種準確率作為因子，以探討準確率的不同將帶給研究數據有何種變化。

2.7 田口品質工程

田口品質工程是由田口玄一博士在 1950 年代為了改善工程品質所提出的方法，其核心思想是以穩健性的設計，降低產品在製造過程中受到外部雜訊或不可控制因數的干擾，達到低成本、高質量的品質工程。

一般來說，品質工程分為線上與線外兩部分：線上品質工程大部分運用改善其產品製程；線外品質工程則是注重於改善其產品製程的設計，對此兩者相較之下，線外品質工程對產品改善較線上品質工程的程度較顯著。田口品質工程屬於後者，並進一步地將其設計分為三個階段[8]：

- (1) 系統設計：以整合當下的技術與系統做為考量，將產能需求盡可能達到最佳化的設計方式
- (2) 參數設計：以更改系統設計參數的方式，得知控制因子最佳化組合，使系統能在雜音因子影響下，依舊能保持其穩健性。
- (3) 公差設計：調整公差範圍的參數，以取得產品品質與產品成本之間的最大利益平衡。

2.8 品質損失函數

田口玄一博士對於品質損失的定義為：當產品出產之後，產生偏離原產品機能之外的變異作用，都稱作品質損失。品質損失越少，代表產品品質越高，與產品機能的目標值越接近；反之，品質損失越高，代表產

品品質越低，與產品機能的目標值越偏遠。根據產品的不同，其目標值、品質特性也不盡相同。其中品質特性分為望大特性、望小特性、望目特性三種[2] [7] [10]。以下介紹品質損失函數[10]如表 2-1 與品質特性表[7]如圖 2-6 所示：

表 2-1 品質損失代號與含意

品質損失代號	含意
L	品質損失
y	品質特性
k	損失係數
m	品質特性目標值
A_0	平均品質損失
Δ_0	消費者容忍限度
$m \pm \Delta_0$	機能極限

其中 $k=A_0/\Delta_0^2$

(1) 望大特性 (Larger The Better, LTB)

產品品質特性 y 越大，代表其品質越佳，損失越小，理想目標值 $m=\infty$ ，其品質損失函數為(2.1)式所示。

$$L(y)=k/y^2 \tag{2.1}$$

(2) 望小特性 (Smaller The Better, STB)

產品品質特性 y 越小，代表其品質越佳，損失越小，理想目標值 $m=0$ ，其品質損失函數為(2.2)式所示。

$$L(y) = ky^2 \quad (2.2)$$

(3) 望目特性 (Nominal The Best, NTB)

產品品質特性 y 越接近 m ，代表其品質越佳，損失越小，理想目標值 $m=y$ ，其品質損失函數為(2.3)式所示。

$$L(y) = k(y - m)^2 \quad (2.3)$$

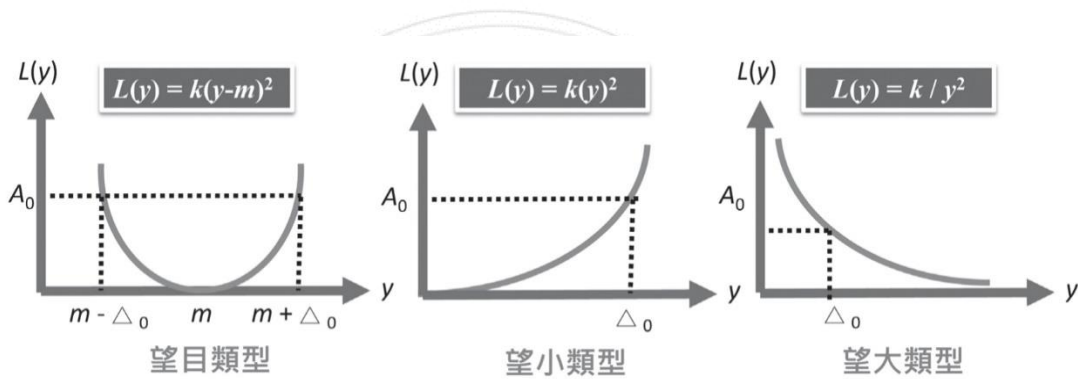


圖 2-6 品質特性表

2.9 訊號雜音比 (Signal-to-Noise, S/N 比)

田口玄一博士利用訊號雜音比設計其品質工程，目的是為了最大化可控制訊號(Signal)，並將不可控制訊號(Noise)最小化，以求達到品質最佳穩健性。由於訊號雜音比源自通訊工程，故以統計方式表達以(2.4)式所示。

$$\eta = \log_{10} \frac{\text{信號}}{\text{雜音}} \quad (2.4)$$

若訊號雜音比越大，表示品質特性平均數與目標值較接近，品質越穩定，其變異數較小，品質損失越少。而訊號雜音比也因品質特性分為望大特性、望小特性、望目特性[2] [9] [10]。表 2-2 為訊號雜音比公式[10]

表 2-2 訊號雜音比公式

品質特性 [↙]	公式 [↙]
望大特性 [↙]	$SN_{LTB} = -10 \log_{10} \left(\frac{1}{n} \sum_{i=1}^n \frac{1}{y_i^2} \right)$ [↙]
望小特性 [↙]	$SN_{STB} = -10 \log_{10} \left(\frac{1}{n} \sum_{i=1}^n y_i^2 \right)$ [↙]
望目特性 [↙]	$SN_{NTB} = 10 \log \left(\frac{\bar{Y}^2}{s^2} \right)$ [↙]

第三章 研究方法

本研究採用卷積神經網路程式為實驗範例，透過田口品質工程的理念，制定流程尋找影響其品質特性之控制因子與變動水準數，依照控制因子計算自由度選擇直交表來排定實驗順序，搭配 Google Colaboratory 軟體執行取得實驗數據，調整控制因子的水準反覆進行實驗，依據實驗結果計算其 S/N 比與品質特性，繪出最佳化因子平均值反應圖與最佳化因子 S/N 比反應圖來解釋各因子對其品質特性其影響，最終求得控制因子最佳化組合。本實驗採取田口實驗方法，其步驟如圖 3-1 所示：



圖 3-1 實驗架構圖

3.1 決定實驗品質特性

本實驗採用 python 機器學習與深度學習特訓班教材卷積神經網路程式為實驗範例，其輸出準確率即是本實驗之品質特性，其品質特性採取望大特性，故輸出準確率越高，代表品質特性越佳。本實驗方法將藉由不同的控制因子的特性與組合，依照田口直交表的排列做量化數據分析，取得最佳化組合數據。

3.2 決定影響品質特性之控制因子與其變動水準數

本實驗以卷積神經網路中的卷積層、池化層、輸出層、拋棄層、隱藏層與訓練方式六個部分做為控制因子，其中隱藏層與訓練方式又各有兩種控制因子，故本實驗共有八個控制因子。以下逐一介紹各個控制因子與其水準數：

(1). 控制因子 A 為輸出層：

採用 Softmax 與 Sigmoid 兩種激勵函數作為水準。

(2). 控制因子 B 為池化層：

採用 Pool_size 設定 (2, 2)、(3, 3)、(4, 4) 三種縮減比例作為水準。

(3). 控制因子 C 為卷積層：

採用 Relu、Elu 與 Tanh 三種激勵函數作為水準。

(4). 控制因子 D 為拋棄層：

在機器學習中，為了防止機器將不必要的樣本特徵作為特徵判斷，而產生的過擬合現象，導致模型準確率下降，故採用拋棄 20%、60%與 80% 三種比例作為水準。

(5). 控制因子 E 為隱藏層 1：

採用 Relu、Elu 與 Tanh 三種激勵函數作為水準。

(6). 控制因子 F 為隱藏層 2：

採用改變神經元的數目為 128、256、512 三種作為水準。

(7). 控制因子 G 為訓練方式 1：

採用 Adam、Adadelata、Adagrad 三種優化器作為水準。

(8). 控制因子 H 為訓練方式 2：

採用 Accuracy、Binary_Accuracy、Categorical_Accuracy 三種準確率作為水準。

依照各個控制因子與水準繪製成圖表，如表 3-1 所示。

表 3-1 實驗品質控制因子與其水準參數

控制因子	代號	水準 1	水準 2	水準 3
輸出層 (激勵函數)	A	Softmax	Sigmoid	
池化層 (Pool_size)	B	(2, 2)	(3, 3)	(4, 4)
卷積層 (激勵函數)	C	Relu	Elu	Tanh
拋棄層 (Dropout)	D	0.2	0.6	0.8
隱藏層 1 (激勵函數)	E	Relu	Elu	Tanh
隱藏層 2 (Units)	F	128	256	512
訓練方式 1 (Optimizer)	G	Adam	Adadelta	Adagrad
訓練方式 2 (Metrics)	H	Accuracy	Binary_Accuracy	Categorical_Accuracy

3.3 計算自由度並套用適合的直交表

田口直交表需透過計算總自由度來選擇實驗所適合的模型，自由度之計算方式採取各項因子的水準數減一，並將所有計算出的結果得總自由度，本實驗總自由度計算方式如下：

$$A \text{ 因子自由度} = 2 - 1 = 1$$

$$B \text{ 因子自由度} = 3 - 1 = 2$$

$$C \text{ 因子自由度} = 3 - 1 = 2$$

$$D \text{ 因子自由度} = 3 - 1 = 2$$

$$E \text{ 因子自由度} = 3 - 1 = 2$$

$$F \text{ 因子自由度} = 3 - 1 = 2$$

$$G \text{ 因子自由度} = 3 - 1 = 2$$

$$H \text{ 因子自由度} = 3 - 1 = 2$$

$$\text{合計總自由度} = 1 + 2 * 7 = 15$$

由於選擇田口直交表時必須注意其行數是否大於總自由度，故本實驗採用 L18($2^1 \times 3^7$)直交表，代表共 18 組實驗，包含 2 水準因子 1 個，3 水準因子 7 個。如表 3-2 所示。

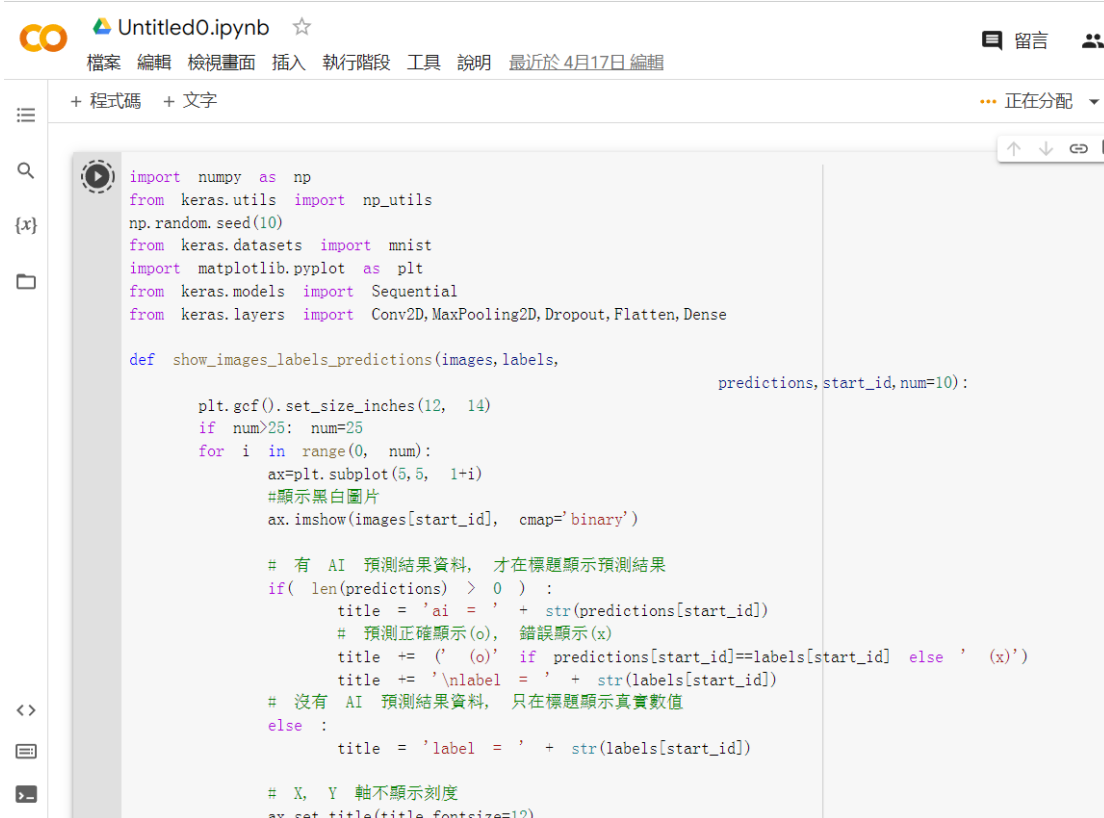
表 3-2 L18 田口直交表

實驗次數\變數因子	A	B	C	D	E	F	G	H
1	1	1	1	1	1	1	1	1
2	1	1	2	2	2	2	2	2
3	1	1	3	3	3	3	3	3
4	1	2	1	1	2	2	3	3
5	1	2	2	2	3	3	1	1
6	1	2	3	3	1	1	2	2
7	1	3	1	2	1	3	2	3
8	1	3	2	3	2	1	3	1
9	1	3	3	1	3	2	1	2
10	2	1	1	3	3	2	2	1
11	2	1	2	1	1	3	3	2
12	2	1	3	2	2	1	1	3
13	2	2	1	2	3	1	3	2
14	2	2	2	3	1	2	1	3
15	2	2	3	1	2	3	2	1
16	2	3	1	3	2	3	1	2
17	2	3	2	1	3	1	2	3
18	2	3	3	2	1	2	3	1

3.4 執行實驗並收集數據

本實驗使用 Google Colaboratory 軟體執行範例程式，透過 L18 直交表的排序，依序執行實驗順序，為了增加數據之準確性，採取量化方式，以每組實驗執行十次，共一百八十組，並將每次實驗數據統計並製作出圖表來分析實驗結果。實驗步驟如下：

(1) 開啟用 Google Colaboratory 軟體，輸入範例程式，如圖 3-1 所示：



```
import numpy as np
from keras.utils import np_utils
np.random.seed(10)
from keras.datasets import mnist
import matplotlib.pyplot as plt
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D, Dropout, Flatten, Dense

def show_images_labels_predictions(images, labels,
                                   predictions, start_id, num=10):
    plt.gcf().set_size_inches(12, 14)
    if num>25: num=25
    for i in range(0, num):
        ax=plt.subplot(5,5, 1+i)
        #顯示黑白圖片
        ax.imshow(images[start_id], cmap='binary')

        # 有 AI 預測結果資料，才在標題顯示預測結果
        if( len(predictions) > 0 ) :
            title = 'ai = ' + str(predictions[start_id])
            # 預測正確顯示(o), 錯誤顯示(x)
            title += (' (o)' if predictions[start_id]==labels[start_id] else ' (x)')
            title += '\nlabel = ' + str(labels[start_id])
        # 沒有 AI 預測結果資料，只在標題顯示真實數值
        else :
            title = 'label = ' + str(labels[start_id])

        # X, Y 軸不顯示刻度
        ax.set_title(title, fontsize=12)
```

圖 3-2 執行實驗畫面(1)

(2) 為加快每次實驗速度，於筆記本設定硬體加速器為 GPU，如圖 3-2 所示：

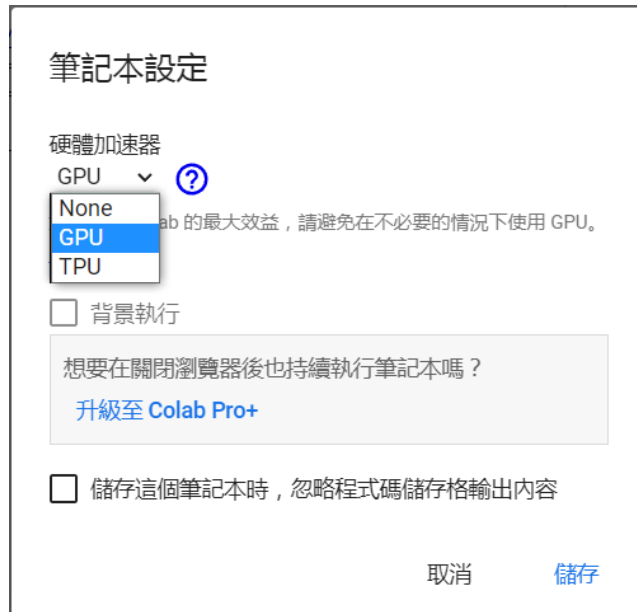


圖 3-3 執行實驗畫面(2)

(3)實驗執行後所得結果如圖 3-3 所示：

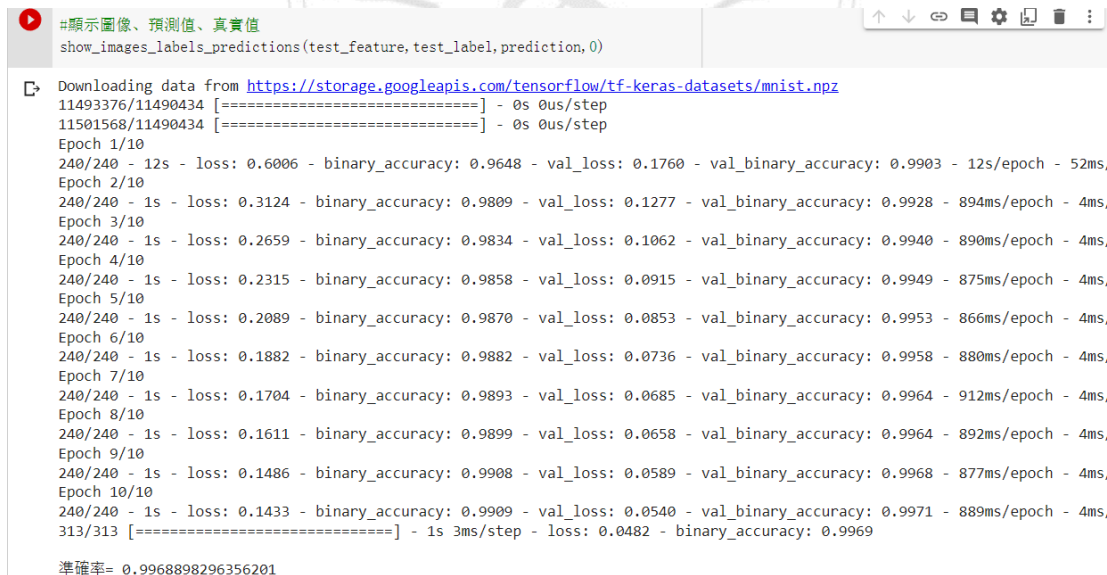


圖 3-4 執行實驗畫面(3)

(4)將一百八十筆實驗結果以 Excel 統計繪製紀錄表。如表 3-3 所示

表 3-3 卷積神經網路實驗準確率紀錄表

組\因子	A	B	C	D	E	F	G	H	Test1	Test2	Test3	Test4	Test5	Test6	Test7	Test8	Test9	Test10
1	1	1	1	1	1	1	1	1	0.9889	0.9887	0.9901	0.9898	0.9882	0.9891	0.9904	0.9899	0.9876	0.9881
2	1	1	2	2	2	2	2	2	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000
3	1	1	3	3	3	3	3	3	0.8614	0.8829	0.8807	0.8854	0.8745	0.8627	0.8726	0.8765	0.8705	0.8628
4	1	2	1	1	2	2	3	3	0.8476	0.8654	0.8759	0.8688	0.8494	0.8511	0.8655	0.8451	0.8319	0.8800
5	1	2	2	2	3	3	1	1	0.9826	0.9850	0.9854	0.9840	0.9836	0.9840	0.9822	0.9845	0.9840	0.9846
6	1	2	3	3	1	1	2	2	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000
7	1	3	1	2	1	3	2	3	0.1322	0.1359	0.0904	0.0975	0.1062	0.1770	0.0988	0.0726	0.1020	0.1136
8	1	3	2	3	2	1	3	1	0.3655	0.2379	0.4246	0.3228	0.3401	0.2808	0.3594	0.3363	0.2910	0.2845
9	1	3	3	1	3	2	1	2	0.9913	0.9918	0.9932	0.9930	0.9919	0.9917	0.9919	0.9917	0.9903	0.9916
10	2	1	1	3	3	2	2	1	0.3816	0.3832	0.3825	0.5122	0.4108	0.4809	0.4581	0.4410	0.5181	0.3308
11	2	1	2	1	1	3	3	2	0.7536	0.7544	0.6574	0.6743	0.6509	0.7039	0.5845	0.5682	0.5747	0.6224
12	2	1	3	2	2	1	1	3	0.9872	0.9880	0.9884	0.9888	0.9865	0.9880	0.9884	0.9875	0.9889	0.9891
13	2	2	1	2	3	1	3	2	0.5489	0.7398	0.7508	0.7158	0.6427	0.4528	0.5738	0.6071	0.6586	0.6833
14	2	2	2	3	1	2	1	3	0.9774	0.9759	0.9760	0.9750	0.9772	0.9744	0.9764	0.9740	0.9784	0.9724
15	2	2	3	1	2	3	2	1	0.3275	0.3448	0.2633	0.3475	0.3383	0.4088	0.3385	0.3170	0.3928	0.3504
16	2	3	1	3	2	3	1	2	0.4996	0.5382	0.5541	0.5394	0.5350	0.5023	0.5004	0.5058	0.5664	
17	2	3	2	1	3	1	2	3	0.0824	0.0965	0.1428	0.0935	0.1067	0.1384	0.1009	0.1413	0.1018	0.0877
18	2	3	3	2	1	2	3	1	0.2150	0.3314	0.2782	0.2691	0.3692	0.3316	0.2268	0.2767	0.2932	0.2181

3.5 依據實驗結果計算 S/N 比、平均值與繪製其反應圖，找出最佳化控制因子組合

本實驗結果採取準確率最佳化，平均值計算方式為每項實驗加總除以該項實驗次數得之，平均值越大代表準確率越高，而 S/N 比則透過望大特性公式計算，S/N 值數值越大代表準確率高。如表 3-4 所示：

表 3-4 S/N 比與平均值計算結果

組\因子	A	B	C	D	E	F	G	H	Test1	Test2	Test3	Test4	Test5	Test6	Test7	Test8	Test9	Test10	SN	平均值
1	1	1	1	1	1	1	1	1	0.9889	0.9887	0.9901	0.9898	0.9882	0.9891	0.9904	0.9899	0.9876	0.9881	-0.09538	0.98908
2	1	1	2	2	2	2	2	2	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	-0.91515	0.90000
3	1	1	3	3	3	3	3	3	0.8614	0.8829	0.8907	0.8854	0.8745	0.8627	0.8726	0.8765	0.8705	0.8625	-1.18118	0.87297
4	1	2	1	1	2	2	3	3	0.8476	0.8654	0.8759	0.8688	0.8494	0.8511	0.8655	0.8451	0.8319	0.8800	-1.33329	0.85807
5	1	2	2	2	3	3	1	1	0.9826	0.9850	0.9854	0.9840	0.9836	0.984	0.9822	0.9846	0.9840	0.9846	-0.1402	0.98399
6	1	2	3	3	1	1	2	2	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	0.9000	-0.91515	0.90000
7	1	3	1	2	1	3	2	3	0.1322	0.1359	0.0904	0.0975	0.1063	0.177	0.0988	0.0726	0.1020	0.1136	-19.6628	0.11262
8	1	3	2	3	2	1	3	1	0.9655	0.9379	0.9246	0.9229	0.9401	0.9209	0.9594	0.9365	0.9210	0.9245	-10.1019	0.92429
9	1	3	3	1	3	2	1	2	0.9910	0.9918	0.9932	0.993	0.9919	0.9917	0.9919	0.9917	0.9909	0.9916	-0.07118	0.99184
10	2	1	1	3	3	2	2	1	0.3816	0.3832	0.3825	0.5122	0.4108	0.4809	0.4881	0.4410	0.5181	0.3308	-7.58492	0.42992
11	2	1	2	1	1	3	3	2	0.7536	0.7544	0.6574	0.6743	0.6509	0.7039	0.5845	0.5682	0.5747	0.6224	-3.80995	0.65443
12	2	1	3	2	2	1	1	3	0.9872	0.9880	0.9884	0.9888	0.9865	0.988	0.9884	0.9875	0.9889	0.9891	-0.10417	0.98808
13	2	2	1	2	3	1	3	2	0.5489	0.7396	0.7508	0.7156	0.6427	0.4526	0.5738	0.6071	0.6556	0.6933	-4.21075	0.63736
14	2	2	2	3	1	2	1	3	0.9774	0.9759	0.9760	0.9750	0.9772	0.9744	0.9764	0.9740	0.9784	0.9724	-0.21362	0.97571
15	2	2	3	1	2	3	2	1	0.3275	0.3448	0.2633	0.3475	0.3383	0.4088	0.3385	0.3170	0.3928	0.3504	-9.46778	0.34289
16	2	3	1	3	2	3	1	2	0.4996	0.5392	0.5541	0.5394	0.5350	0.5023	0.5028	0.5004	0.5058	0.5664	-5.63304	0.52440
17	2	3	2	1	3	1	2	3	0.0924	0.0965	0.1429	0.0935	0.1067	0.1384	0.1009	0.1413	0.1018	0.0877	-19.6986	0.10920
18	2	3	3	2	1	2	3	1	0.2150	0.3314	0.2782	0.2691	0.3692	0.3316	0.2268	0.2767	0.2392	0.2181	-11.4319	0.28093

利用 S/N 比與平均值計算結果，以每一因子與各水準為單位，按照直交表的排序將其 S/N 比或平均值加總後再取平均值，依序求得最佳化因子 S/N 反應表與最佳化因子平均值反應表，詳見表 3-5 與表 3-6，爾後再以反應表繪製出最佳化因子 S/N 反應圖與最佳化因子平均值反應圖，如圖 3-5 與圖 3-6 所示：

表 3-5 最佳化因子 S/N 反應表

水準\因子	A	B	C	D	E	F	G	H
1	-3.82402	-2.28179	-6.42002	-5.74603	-6.02146	-5.85433	-1.04293	-6.47035
2	-6.90609	-2.71347	-5.81324	-6.07749	-4.59255	-3.59168	-9.7074	-2.59254
3		-11.0999	-3.86189	-4.27163	-5.48114	-6.64915	-5.34483	-7.03227

表 3-6 最佳化因子平均值反應表

水準\因子	A	B	C	D	E	F	G	H
1	0.770318	0.805747	0.591908	0.657585	0.652128	0.658002	0.90885	0.558517
2	0.549213	0.783003	0.657937	0.650497	0.656288	0.739412	0.465772	0.768005
3		0.390547	0.729452	0.671215	0.67088	0.581883	0.604675	0.652775

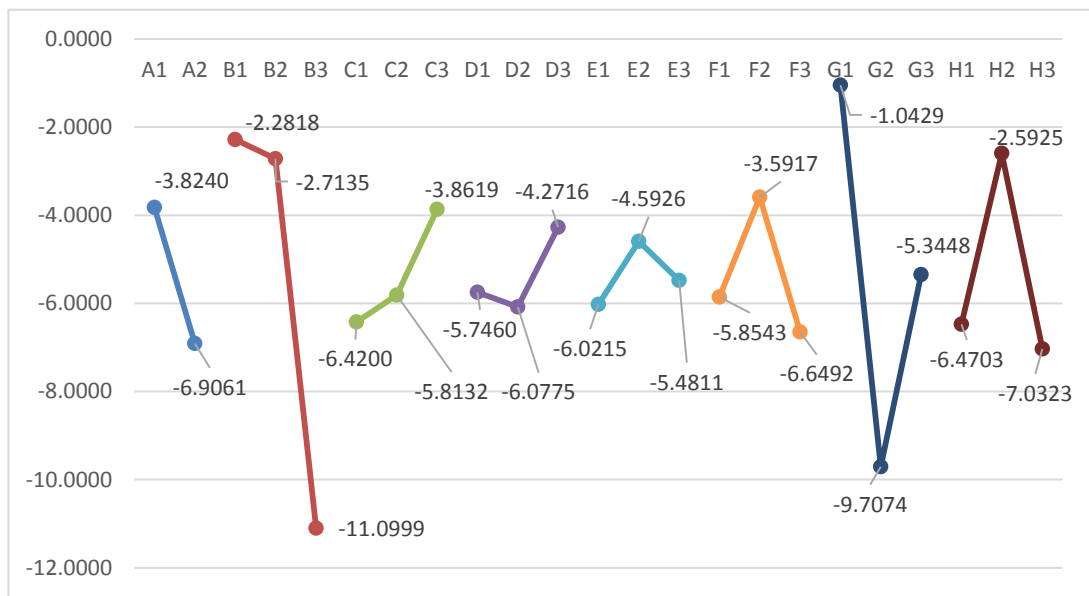


圖 3-5 最佳化因子 S/N 反應圖

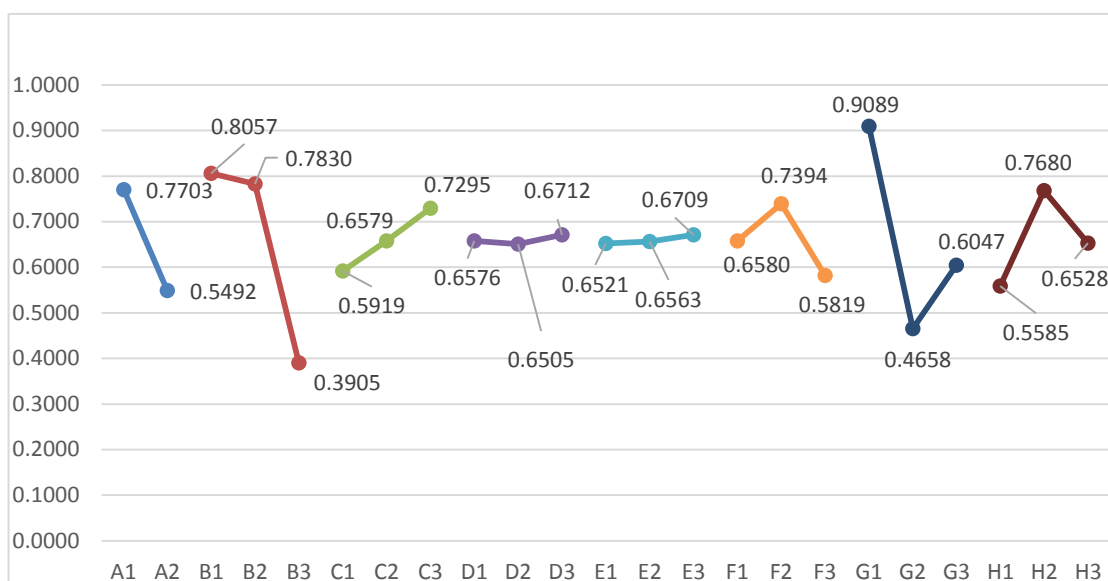


圖 3-6 最佳化因子平均值反應圖

由最佳化因子 S/N 反應圖與最佳化因子平均值反應圖兩圖對照相比較下，最佳化因子平均值反應圖對於控制因子 D、E 兩因子差異較不顯著，而最佳化因子 S/N 反應圖控制因子 D、E 兩因子較為顯著差異，故採用最佳化因子 S/N 反應圖作為最佳化因子選取，其最佳化組合為 $A_1B_1C_3D_3E_2F_2G_1H_2$ 。

第四章 實驗結果分析

經實驗結果進行最佳化控制因子組合確認實驗。以本實驗原範例程式的控制因子採用 $A_1B_1C_1D_1E_1F_2G_1H_1$ ，輸出層激勵函數採用 Softmax，池化層為 Pool_size(2, 2)，卷積層為 Relu，拋棄層為 0.2，隱藏層激勵函數採用 Relu，隱藏層神經元為 256，訓練方式優化器為 Adam，訓練方式準確率模型為 Accuracy。以最佳因子預測式計算最初 S/N 比與平均值。

$$\begin{aligned}
 S/N_{initial} &= \bar{\eta} + (\bar{\eta}_{A_1} - \bar{\eta}) + (\bar{\eta}_{B_1} - \bar{\eta}) + (\bar{\eta}_{C_1} - \bar{\eta}) + (\bar{\eta}_{D_1} - \bar{\eta}) + (\bar{\eta}_{E_1} - \bar{\eta}) + (\bar{\eta}_{F_2} - \bar{\eta}) + (\bar{\eta}_{G_1} - \bar{\eta}) \\
 &\quad + (\bar{\eta}_{H_1} - \bar{\eta}) \\
 &= \bar{\eta}_{A_1} + \bar{\eta}_{B_1} + \bar{\eta}_{C_1} + \bar{\eta}_{D_1} + \bar{\eta}_{E_1} + \bar{\eta}_{F_2} + \bar{\eta}_{G_1} + \bar{\eta}_{H_1} - 7\bar{\eta} \\
 &= -3.82402 - 2.28179 - 6.42002 - 5.74603 - 6.02146 - 3.59168 - 1.04293 - 6.47035 \\
 &\quad - 7(-5.36505) \\
 &= 2.157076(\text{db})
 \end{aligned}$$

$$\begin{aligned}
 \mu_{initial} &= \bar{\mu} + (\bar{\mu}_{A_1} - \bar{\mu}) + (\bar{\mu}_{B_1} - \bar{\mu}) + (\bar{\mu}_{C_1} - \bar{\mu}) + (\bar{\mu}_{D_1} - \bar{\mu}) + (\bar{\mu}_{E_1} - \bar{\mu}) + (\bar{\mu}_{F_2} - \bar{\mu}) + (\bar{\mu}_{G_1} - \bar{\mu}) \\
 &\quad + (\bar{\mu}_{H_1} - \bar{\mu}) \\
 &= \bar{\mu}_{A_1} + \bar{\mu}_{B_1} + \bar{\mu}_{C_1} + \bar{\mu}_{D_1} + \bar{\mu}_{E_1} + \bar{\mu}_{F_2} + \bar{\mu}_{G_1} + \bar{\mu}_{H_1} - 7\bar{\mu} \\
 &= 0.770318 + 0.805747 + 0.591908 + 0.657585 + 0.652128 + 0.739412 + 0.90885 \\
 &\quad + 0.558517 - 7 * 0.659766 \\
 &= 1.066106
 \end{aligned}$$

本實驗結果的最佳控制因子採用 $A_1B_1C_3D_3E_2F_2G_1H_2$ ，輸出層激勵函數採用 Softmax，池化層為 Pool_size(2, 2)，卷積層為 Tanh，拋棄層為 0.8，隱藏層激勵函數採用 Elu，隱藏層神經元為 256，訓練方

式優化器為 Adam，訓練方式準確率模型為 Binary_Accuracy。以最佳

因子預測式計算最佳 S/N 比與平均值。

$$\begin{aligned}
 S/N_{\text{Optimizer}} &= \bar{\eta} + (\bar{\eta}_{A1} - \bar{\eta}) + (\bar{\eta}_{B1} - \bar{\eta}) + (\bar{\eta}_{C3} - \bar{\eta}) + (\bar{\eta}_{D3} - \bar{\eta}) + (\bar{\eta}_{E2} - \bar{\eta}) + (\bar{\eta}_{F2} - \bar{\eta}) + (\bar{\eta}_{G1} - \bar{\eta}) \\
 &\quad + (\bar{\eta}_{H2} - \bar{\eta}) \\
 &= \bar{\eta}_{A1} + \bar{\eta}_{B1} + \bar{\eta}_{C3} + \bar{\eta}_{D3} + \bar{\eta}_{E2} + \bar{\eta}_{F2} + \bar{\eta}_{G1} + \bar{\eta}_{H1} - 7\bar{\eta} \\
 &= -3.82402 - 2.28179 - 3.86189 - 4.27163 - 4.59255 - 3.59168 - 1.04293 \\
 &\quad - 2.59254 - 7 * (-5.36505) \\
 &= 11.49633(\text{db})
 \end{aligned}$$

$$\begin{aligned}
 \mu_{\text{Optimizer}} &= \bar{\mu} + (\bar{\mu}_{A1} - \bar{\mu}) + (\bar{\mu}_{B1} - \bar{\mu}) + (\bar{\mu}_{C3} - \bar{\mu}) + (\bar{\mu}_{D3} - \bar{\mu}) + (\bar{\mu}_{E2} - \bar{\mu}) + (\bar{\mu}_{F2} - \bar{\mu}) + (\bar{\mu}_{G1} - \bar{\mu}) \\
 &\quad + (\bar{\mu}_{H1} - \bar{\mu}) \\
 &= \bar{\mu}_{A1} + \bar{\mu}_{B1} + \bar{\mu}_{C3} + \bar{\mu}_{D3} + \bar{\mu}_{E2} + \bar{\mu}_{F2} + \bar{\mu}_{G1} + \bar{\mu}_{H1} - 7\bar{\mu} \\
 &= 0.770318 + 0.805747 + 0.729452 + 0.671215 + 0.656288 + 0.739412 + 0.90885 \\
 &\quad + 0.768005 \\
 &= 1.430927
 \end{aligned}$$

將最佳化因子組合與原範例程式的控制因子組合兩者皆以最佳

化因子預測式計算結果相比：

$$S/N \text{ 比增益差} : 11.49633 - 2.157076 = 9.339254(\text{db})$$

$$\text{平均值差} : 1.430927 - 1.066106 = 0.364821$$

得知本實驗最佳化因子組合可提高原現行因子組合之準確率，同時提

升卷積神經網路之辨識率。

確認實驗主要針對最佳化因子組合再次以實驗方式去得知其準確率，得知後的結果與直交表實驗中最佳準確率第九組因子組合相比，進一步確認何者為最佳化因子組合。最佳化因子實驗數據如表 4-1 所示：

表 4-1 最佳化因子實驗數據

Test1	Test2	Test3	Test4	Test5	Test6	Test7	Test8	Test9	Test10	SN	平均值
0.9968	0.9972	0.9971	0.9973	0.9972	0.9973	0.9973	0.9971	0.9965	0.9971	-0.02531	0.99709

直交表實驗中數據以第九組因子組合準確率為最佳，相比最佳化因子組合之下，最佳化因子組合在S/N比與平均值皆優異於第九組因子組合，故可再次確認此組合乃是最佳化因子組合，且以十次實驗數據相比得知，準確率數值差異不大，故得知此最佳化因子組合也同時具備穩定性。

第五章 結論

文字辨識是機器學習發展以來，一直不斷追求進步的技術之一，早期辨識技術大多是屬於監督式學習，透過大量的資料註記，才能提高機器辨識的準確度，如此一來將會提高人力成本，且因為早期機器學習的演算法大多是屬於線性計算，而現實中的辨識物大多屬於非線性，故早期的機器辨識準確度一直無法提升，自深度學習卷積神經網路演算法問世之後，文字辨識技術大幅提升，發展至今文字辨識的方式也越發普遍，目前以光學自元辨識 OCR 最為主流，各種 OCR 辨識軟體如雨後春筍般的開發出來，就連 Google 與通訊軟體 Line 都加入其中行列，其操作簡單也方便。而在各種辨識軟體競爭下，若以消費者角度來從中選擇其一，軟體的辨識度準確率將是最為重要的一環。

軟體準確率主要來自於卷積神經網路演算法的設計為主，本研究選擇透過田口品質工程的理念，設立直交表，找出控制因子激勵函數最佳化組合，藉此提升卷積神經網路演算法準確率。為了證明最佳化因子組合，本實驗最後以最佳化因子組合結果交叉比對最初程式控制因子組合與直交表第九組控制因子組合兩次，結果最佳化因子組合皆優於兩者，再次確認田口實驗方式確實能尋找出最佳化控制因子與透過實驗結果的最佳

化組合確實能提升卷積神經網路演算法之準確率，具有參考實用性。



參考文獻

- [1] 洪仁德(2021)。基於卷積神經網路影像識別之草藥 辨識系統之設計。明新科技大學電機工程碩士論文，新竹縣。
- [2] 劉邦彥(2021)。類自主機器學習與卷積加速器之研究與合成。成功大學電機工程研究所碩士論文，台南市。
- [3] 胡依淳(2008)。深度卷積神經網路中卷積層之分析及比較。暨南國際大學電機工程學系碩士論文，南投縣。
- [4] 黃梓鉞(2021)。基於卷積神經網路影像識別之草藥 辨識系統之設計。明新科技大學電機工程碩士論文，新竹縣。
- [5] 范慶龍(2021)。監督式機器學習於土地覆蓋分類效益之研究。台灣土地研究 24 卷 1 期，P67-94。
- [6] 黃成凱、黃緒哲、嚴健榮、古有彬、楊淳宜、陳俊浩(2018)。基於深度學習之工件自動辨識與取料之應用。機械工業雜誌 428 期，P85-95。
- [7] 韓文銘、廖珮玟(2012)。整合田口損失函數與層級分析法於軟體產品使用者品質評估之應用。品質學報 19 卷 2 期，P97-116。
- [8] 余豐榮、莊銘國、施啟順(2007)。以田口方法提升鑄件品質之

- 研究。鑄造工程學刊 33 卷 4 期，P17-24。
- [9] 李友錚(2007)。重要度與滿意度分析：田口訊號雜音比解析法。
品質月刊 43 卷 5 期，P68-73、P72-76。
- [10] 陳彥旗(2021)。應用田口法於杉林溪高山茶之輕焙火最佳化
研究。雲林科技大學工業工程與管理系碩士論文，雲林縣。
- [11] 洪偉嘉(2019)。應用卷積類神經網路於中文簽名圖形辨識之研
究。技術學刊 34 卷 3 期，P145-157。
- [12] 林春妙 (2012)。應用田口方法於咖啡沖製最佳化。南華大學資
訊管理系碩士論文，嘉義縣。
- [13] 葉大銘 (2017)。利用田口分析法對數位乳房假體影像品質的最
佳化與臨床驗證。中臺科技大學醫學影像暨放射科學系暨研究所博
士論文，台中市。
- [14] 鄧文淵 (2021)。Python 機器學習與深度學習特訓班 (第二版):
看得懂也會做的 AI 人工智慧實戰。基峯資訊出版。
- [15] 陳政琳 (2020)。以田口實驗法找出降低表面粗糙度的最佳化參數。
逢甲大學資訊電機工程碩士論文，台中市。
- [16] 屈文華 (2014)。應用田口方法分析製程最佳化之研究。社會服務與
休閒產業研究 3 期，P46-61。

- [17] Hafemann, L. G., R. Sabourin, and L. S. Oliveira, L. S. (2016). *Analyzing Features Learned for Offline Signature Verification Using Deep CNNs*. In Proceedings of the 23rd International Conference on Pattern Recognition, Cancun, Mexico.
- [18] Hafemann, L. G., R. Sabourin, and L. S. Oliveira. (2017). *Learning Features for Offline Handwritten Signature Verification Using Deep Convolutional Neural Networks*. Pattern Recognition 70: 163-176.
- [19] Bhandare, A., M. Bhide, P. Gokhale, and R. Chandavarkar . (2016). *Applications of Convolutional Neural Networks*. International Journal of Computer Science and Information Technologies 7 (5): 2206-2215.
- [20] Glorot, X., and Y. Bengio . (2010). *Understanding the Difficulty of Training Deep Feedforward Neural Networks*. Proceedings of Journal of Machine Learning Research (JMLR) W&CP 9 (9): 249-256.
- [21] Huang, G., Z. Liu, L. v. d. Maaten, and K. Q. Weinberger.(2017). *Densely Connected Convolutional Networks*. 2017 IEEE Conference on Computer Vision and Pattern Recognition, Hawaii, 21-26.
- [22] R. S. Chen, H. H. Lee and C. Y. Yu. (1997). *Application of Taguchi's Method on the Optimal Process Design of an Injection Molded PC/PBT Automobile Bumper*, Composite Structures, Vol.39, pp209-214.