論文題目：視覺化模擬輔助類神經網路教學之研究

The Effects of Visualization and Simulation on Learning of Neural Networks

第一章 緒論

　　本章共分為三節，第一節說明本研究之背景與動機以及為何使用模擬式教學輔助人工智慧概念與實作的學習；第二節提出具體的研究目的；第三節則為本研究中所提及之重要名詞的名詞釋義。

第一節 研究背景與動機

　　當今社會中，電腦科學與人工智慧的科技應用深入我們的日常生活。人工智慧的主要研究主題(例：機器學習、深度學習)，其能夠影響我們的生活面向甚廣，像是居家機器人、交通、健康照護、教育、公共安全、職場，甚至是娛樂(Stone et al., 2016)。我國教育部也將人工智慧的概念納入108課綱中(教育部，2018)，國際上也有教育相關政策將人工智慧相關議題納入探討，像是美國 CSTA 2017 年的課綱中，針對 11 和 12 年級的學習年段，也列入人工智慧演算法的實作(CSTA, 2017)，聯合國教科文組織於2019年討論如何在教育環境中教導人工智慧(Pedro, Subosa, Rivas, & Valverde, 2019)。

　　由於人工智慧的普及，有些學者認為孩子除了具備傳統素養(例：讀寫素養、數學素養)，孩子對於人工智慧的素養也同等重要，應思考如何從小教導孩子認識人工智慧 (Kandlhofer et al., 2016; Burgsteiner et al., 2016)。從國際上到國內教育趨勢，可知思考如何教導人工智慧這項學科知識是很重要的議題。

　　然而，目前人工智慧教育相關研究，教學對象多數為大學或研究所學生(Barella, Valero, & Carrascosa, 2008; Cuéllar and Pegalajar, 2014; Fernandes, 2016; Grivokostopoulou and Hatzilygeroudis, 2014; Kochlán and Hodon, 2014; Kumar, 2004; Marković, Kostić Kovačević, Nikolić, & Nikolić, 2015; Yoon and Kim, 2015)，這對於國高中、國小階段開始有人工智慧領域的教育需求來說，沒有相關教學經驗與教學研究，在制定或發展相關教材上就會缺乏參考方針，而且這些人工智慧相關的教學研究，並沒有設計教學實驗以探討何種教學方法對學生的學習是有顯著影響的，這可能都會讓國高中、國小階段的教師缺乏理解應該使用何種方式教導學生。而人工智慧為電腦科學重要領域，也與程式設計、演算法的學習息息相關(Jacob and Warschauer, 2018)，我們若要知道此學科的教學困難，以及適當的教學方法，則應該參考過去在電腦科學或程式設計、演算法教學上有何種教學困難，了解這些困難，才得以發展合適的教材供學生學習人工智慧相關知識。

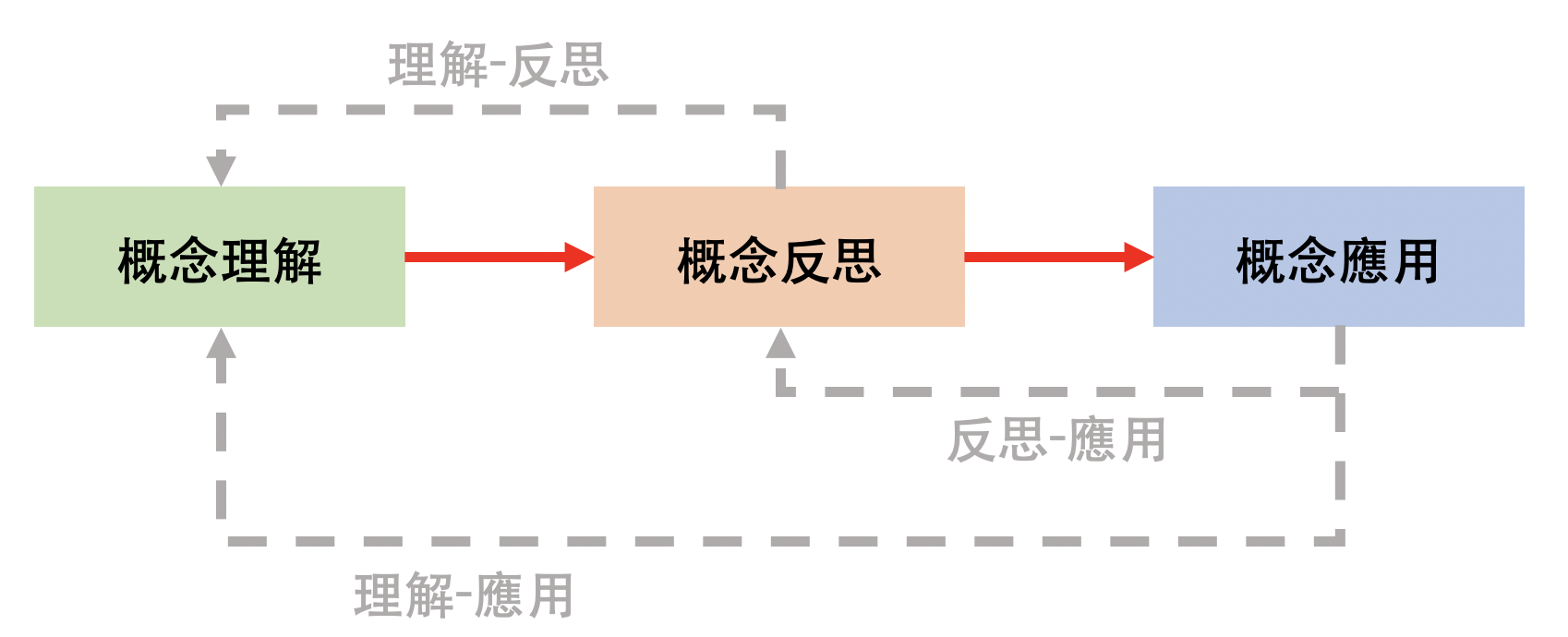
　　程式設計和演算法被認為不容易學習的原因在於，普遍學生在學習「抽象概念」時，難以將抽象概念與現實生活形成連結，進而造成學生不知道如何運用程式來撰寫問題解決的方法(Esteves, Fonseca, Morgado, & Martins, 2011; Bellstrom and Thoren, 2009)。而先前研究了解到，讓學生先理解演算法對此類知識的學習是有幫助的，製作演算法的動畫以及互動式的教學內容，也能夠讓學生更佳理解演算法的主要概念(Végh & Stoffová, 2017)。所以本研究將使用演算法視覺化與模擬式教學的方式教導人工智慧。

　　演算法視覺化(Algorithm Visualization)能夠透過圖像化的方式呈現演算法執行過程中的動態變化，比起虛擬碼或靜態圖表，更能夠幫助學生理解演算法的抽象概念(Simoňák, 2016; Tudoreanu, Wu, Hamilton-Taylor, & Kraemer, 2002)。

　　模擬式教學(Simulation Learning)被普遍使用在科學教育研究中(Jensen et al., 2002; Khoo & Koh, 1998; Luo et al., 2005)，而且依照Alessi和Trollip(2001)或De Jong 和Van Joolingen(1998)對於模擬式教學的定義，在電腦科學教育中，演算法視覺化輔助教學的研究，也屬於模擬式教學的一種，即便這些研究並不一定使用模擬式教學的詞彙描述研究中使用的教學方式(Abu-Naser, 2008; Colaso et al., 2002; Korhonen & Malmi, 2000; McNally, Naps, Furcy, Grissom, & Trefftz, 2007; Naps et al., 2003; Simoňák, 2016; Tudoreanu et al., 2002; Végh & Stoffová, 2017)，在本研究後續的敘述，為了容納更廣泛的研究領域與研究觀點，我們將用模擬式教學指稱本研究製作的教學教材。然而，不論是模擬式教學或演算法視覺化的相關研究中，都有研究得出一個共同結論說明這類型的教學方法是因為提升學生的操作和互動機會而增進學習(Colaso et al., 2002; Jensen, Self, Rhymer, Wood, & Bowe, 2002; Korhonen & Malmi, 2000; Naps et al., 2003; Saraiya, Shaffer, McCrickard, & North, 2004; Tversky, Morrison, & Betrancourt, 2002)，而且這樣的結論與一些學習理論的說法一致，互動式的學習活動能夠引起動機，進而增進學習成效(Prensky, 2002)。在Chen, Hong, Sung和Chang(2011)的研究中，也說明到模擬式教學，讓學生經歷概念學習(concept learning)、模擬操作(simulative manipulation)、概念澄清(concept clarification)，讓學生透過參數的調整學習到概念，有基礎概念後，在模擬工具上做到更詳細的概念澄清，在這過程中，學生可以經歷反思的循環。而且透過科技工具輔助學習時，要促使學習者進行「有意義的學習(meaningful learning)」，教材或教學活動需要滿足幾個精神，包含「建構」、「溝通」、「表達」、「合作」、「真實」、「反思」等(Jonassen et al., 2000)，而本研究認為模擬式教學，透過模擬工具，讓學生能夠真實地操作一個概念，進而更深層地建構知識，在課堂討論時，也更容易溝通與表達在課堂所學習到的知識。

　　而演算法視覺化的長期研究也理解出使這類教材有效的因素(Shaffer et al., 2010)，像是給予學生控制演算法的執行步驟，讓學生更深入地理解演算法(Saraiya et al., 2004)，或是將演算法的執行過程比喻成日常的生活經驗，並透過動畫呈現，更能幫助學生理解演算法的概念(Hansen, Narayanan, & Schrimpsher, 2000)。

　　依照上述人工智慧教育的趨勢，以及此學科在國高中階段的教學研究稀缺，還有模擬式教學與演算法視覺化的特色與過去的研究成果證實此教學方法能夠讓此學科的教學更有效果，本研究設計模擬式教學策略與視覺化模擬輔助教學平台，搭配人工智慧教材教導高中學生學習人工智慧領域中的類神經網路演算法。本研究的模擬式教學策略中，包含「概念理解」、「概念反思」、「概念應用」三項教學流程(如圖一)。



圖一、本研究模擬式教學策略

　　至於教學策略中的流程如何進行，以及教學平台有哪些內容與互動的功能，本研究將在研究方法的章節中詳細說明。

第二節 研究目的

一、探究視覺化模擬輔助教學對人工智慧學習成就之影響。

　　1. 對於人工智慧概念之影響。

　　2. 對於人工智慧演算法實作之影響。

二、探究視覺化模擬輔助教學對人工智慧學習態度之影響。

　　1. 電腦科學學習動機

　　2. 電腦科學自我效能

　　3. 資訊科學抽象概念/程序之學習感受

　　4. 對於人工智慧學習自我評鑑

三、探究模擬式教學策略的課堂感受

　　1. 對「概念理解」過程的課堂感受

　　2. 對「概念反思」過程的課堂感受

　　3. 對「概念應用」過程的課堂感受

　　4. 「概念理解」、「概念反思」、「概念應用」課堂感受之比較

第三節 名詞釋義

第二章 文獻探討

　　本章針對研究主題以及相關理論進行探討，共分為四節，第一節為人工智慧，探討人工智慧教育的趨勢以及人工智慧教學相關研究；第二節為程式設計與演算法教學，說明程式設計與演算法的教學問題，以及運算思維與其關聯；第三節為模擬式教學，探討此教學方法的特色與其應用於不同學科(例如：電腦科學、自然科學)的研究成果；第四節為演算法視覺化，探討此教學方法如何幫助學生學習演算法以及其相關研究成果。

第一節 人工智慧

一、人工智慧教育的趨勢

　　現今人工智慧的主要研究主題(例：機器學習、深度學習、機器人、電腦視覺、物聯網、自然語言處理)，其能夠影響我們的生活面向甚廣，像是居家機器人、交通、健康照護、教育、公共安全、職場，甚至是娛樂(Stone et al., 2016)。人工智慧這項科技，能夠嵌入於硬體的機械上，也能夠應用於軟體的運作與開發上，人們經常使用的行動裝置，及其提供的應用與服務，皆有可能是基於人工智慧所開發的，例如：Google、Siri、Cortana、自駕車、聊天機器人(Kandlhofer, Steinbauer, Hirschmugl-Gaisch, & Huber, 2016; Burgsteiner, Kandlhofer, & Steinbauer, 2016; Estevez, Garate, Guede, & Grana, 2019)。

　　我們所處的生活環境，食衣住行育樂再也難以與人工智慧科技分離，所以我們應該培養孩子擁有足夠的能力理解這個充斥著人工智慧的社會，我國教育部也將人工智慧的概念納入108課綱中(教育部，2018)，也針對國內中小學設計人工智慧相關教材《和 AI 做朋友》(黃仁暐、涂益郎，2019），在美國 CSTA 2017 年的課綱中，針對 11 和 12 年級的學習年段，也列入人工智慧演算法的實作(CSTA, 2017)，聯合國教科文組織於2019年也討論人工智慧在教育領域中的挑戰與機會，探討人工智慧如何影響教育，重新思考如何培養學生擁有程式設計能力以面對現今社會(Pedro, Subosa, Rivas, & Valverde, 2019)。可見在國際上，人工智慧與教育的連結實屬重要，不論是在生活面向或教育政策面向，我們都應該重視人工智慧這項議題。

　　然而，由於人工智慧已經與我們的社會密不可分，也有學者認為孩子除了具備傳統素養(例：讀寫素養、數學素養)，孩子對於人工智慧的素養也同等重要，應思考如何從小教導孩子認識人工智慧，並且根據Russell 和 Norvig (2002)所撰寫的人工智慧教科書，訂定人工智慧素養，其中包含「自動化」、「智慧代理」、「圖論與資料結構、電腦前學基礎知識」、「排序」、「透過搜尋解決問題」、「經典規劃」、「機器學習」 (Kandlhofer et al., 2016; Burgsteiner et al., 2016)，訂定出明確的人工智慧素養，也使教學或研究更容易界定何種教學主題屬於人工智慧教育。

　　另外，由於人工智慧相關知識與電腦科學息息相關，近期電腦科學教育領域討論的運算思維，也與人工智慧的知識有所關聯(Jacob and Warschauer, 2018)，所以本研究除了回顧人工智慧教育相關研究，也同時參考了運算思維與程式教育相關研究以發展此教材，其內容會在後續的章節說明。

二、人工智慧教育的相關研究

　　目前人工智慧教育相關研究，教學對象多數為大學或研究所學生(Barella, Valero, & Carrascosa, 2008; Cuéllar and Pegalajar, 2014; Fernandes, 2016; Grivokostopoulou and Hatzilygeroudis, 2014; Kochlán and Hodon, 2014; Kumar, 2004; Marković, Kostić Kovačević, Nikolić, & Nikolić, 2015; Yoon and Kim, 2015)，這對於國高中、國小階段開始有人工智慧領域的教育需求來說，沒有相關教學經驗與教學研究，在制定或發展相關教材上就會缺乏參考方針，而且在大學階段的教學研究，很有可能學生已經具備一定程度的學科專業知識(例：Cuéllar和Pegalajar(2014)教導的學生為電腦工程科系大學三年級的學生)，與國高中、國小階段的學生有相當大的學科能力差距，以至於這些教學案例或研究，對於此學習階段的參考價值並不高。

　　而且大學階段以上的教學研究，有的教學研究僅僅介紹了課程架構、內容，或是課程進行的方式，並沒有設計教學實驗進行研究，像是Cuéllar和Pegalajar(2014)使用LEGO機器人發展教材來教導機器人程式設計、規劃演算法、審議型代理、搜尋演算法等等教學主題，有進行實際教學，但對於實際上解決什麼教學問題，論文當中沒有明確寫出，而是在硬體設備和專案的內容有很詳細的交代，而Yoon和Kim(2015)則是透過憤怒鳥遊戲發展人工智慧教材，利用遊戲式學習與專案導向式學習策略，教導學生策略決策制定、電腦視覺、資源管理等主題；有的研究則是介紹教導人工智慧課程適合使用的硬體，或是軟體環境，像是Barella, Valero和 Carrascosa(2008) 開發一個多智慧體系統(Multiagent System)，讓學生能夠在這個系統上開發自己的智慧代理人，並且加入奪旗遊戲(Capture-the-Flag)的機制，讓學生們開發的智慧代理人能夠互相競爭，而Grivokostopoulou和Hatzilygeroudis(2014)則是透過語意網路(Semantic Web)發展個人化學習的教材來教導人工智慧相關主題，類似的研究還有Marković等人(2015)發展智慧型系統讓學生進行自主學習人工智慧相關知識，Kochlán和Hodon(2014)介紹Yrobot這個開放式硬體的架構與功能，並說明此硬體能夠發展人工智慧教材。上述在大學階段的教學研究，雖然提供了人工智慧教學主題的實際案例或方法，但皆無採取分組教學的研究方法，這樣將難以檢驗何種教學設計實際影響了學生的學習成效，並且也只有少數研究確實說明人工智慧教學的疑問(例如：機器人專案對於學生習得人工智慧概念是否有效率？)且試圖使用教學數據回答這些問題(Kumar, 2004)。

　　高中階段的教學研究則有探討學生參與機器人競賽的影響(Sklar, Eguchi, & Johnson, 2002)，但其中也只有調查學生對於參與競賽的感受，並沒有對於學生的學科能力改變進行調查。

　　即便現有人工智慧素養相關研究的出現，讓幼兒階段，以及國中、國小階段有可行的人工智慧課程，但其相關研究的評估方法也僅旨在確認有達成研究自身訂定的教學目標(Kandlhofer et al., 2016; Burgsteiner et al., 2016)，同樣沒有透過教學數據驗證何種教學設計影響著學生的學習成效。

　　然而，先前人工智慧教學研究較少提及此教學主題上可能會遇到的困難(Barella, Valero, & Carrascosa, 2008; Cuéllar and Pegalajar, 2014; Fernandes, 2016; Grivokostopoulou and Hatzilygeroudis, 2014; Kochlán and Hodon, 2014; Kumar, 2004; Marković et al., 2015; Yoon and Kim, 2015)，這也可能因為人工智慧教學為現代的新興教學主題，所以較無前人整理此學科的教學問題與困境。既然人工智慧這個學科屬於電腦科學，也與程式設計、演算法的學習息息相關(Jacob and Warschauer, 2018)，所以本研究認為應該要參考相關學科的教學研究，藉此了解人工智慧教學可能會遇到的困難，知道教學上的困難，才得以運用合適的教學方法解決這些困難。

　　在第二節「程式設計與演算法教學」，我們將會透過探討程式與演算法教學上的困難，思考人工智慧教學中也可能遇到的困難，再透過第三節、第四節介紹「模擬式教學」與「演算法視覺化」的特色發展教材以克服這些教學困難，並且以教學實驗的方式檢驗這樣的教學設計對學習有實質幫助。

第二節 程式設計與演算法教學

一、運算思維

　　隨著資訊科技的進步，培養學生問題解決能力、運算思維能力，程式設計與演算法成為二十一世紀重要的學習科目，在人工智慧素養被提出之前(Kandlhofer et al., 2016; Burgsteiner et al., 2016)，已有學者提出運算思維的概念以及其重要性，運算思維(Computational thinking)被描述為人類思考問題解決的一種方式，利用電腦科學的基本概念進行問題解決的思維(Wing, 2006)。如果學生擁有運算思維，當學生在面對問題時，就有機會運用電腦科學家的思維進行問題解決(Selby & Woollard, 2013)，在這日常生活充斥著科技的時代，學生擁有運算思維才有能力善用或掌握科技。Selby和 Woollard(2013)認為運算思維著重問題解決的過程，擁有抽象(Abstraction)、分解(Decomposition)、演算法設計(Algorithmic design)、評估(Evaluation)、一般化(Generalization)的思維過程。

　　透過學習程式設計與演算法，學生能夠更加貼近電腦科學家的思維方式，並且因為運算思維、程式設計與演算法、人工智慧，本身就是電腦科學領域的相關知識(Jacob and Warschauer, 2018)，所以本研究抱持著程式設計與演算法是學生認識人工智慧的基石，以及培養運算思維的途徑之一，了解過去相關教學研究，將更能發展合適的教材教導學生人工智慧，並且培養學生運算思維的能力。

二、程式設計與演算法教學的困難

　　程式設計和演算法被認為不容易學習的原因在於，普遍學生在學習「抽象概念」時，難以將抽象概念與現實生活形成連結，進而造成學生不知道如何運用程式來撰寫問題解決的方法(Esteves, Fonseca, Morgado, & Martins, 2011)。

　　學生在學習程式設計或演算法時，通常需要四種基礎知識才能熟悉程式設計的技巧(Bellstrom and Thoren, 2009)：

* 解決問題的基礎數學知識。
* 了解如何操作撰寫程式的環境(programming environment)的知識，例如：如何編譯與執行程式碼？如何進行程式偵錯？
* 程式語言相關知識，例如：資料結構、函式，或是特定程式語言的語法。
* 將整體學習到的知識轉化成程式邏輯的知識，例如：使用特定的資料結構、函式、控制流程做特定問題的解決方法。

　　然而，第四種知識對於學生而言相對困難，製作演算法的動畫以及互動式的教學內容是能夠有效解決這個困難的 (Végh & Stoffová, 2017)，因為先前研究發現動畫與互動式的教材能夠幫助學生學習程式設計與演算法，所以本研究透過模擬與演算法視覺化的方式來輔助學生學習人工智慧。

第三節 模擬式教學

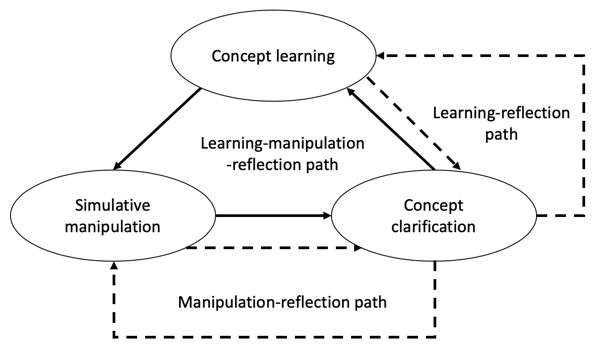
一、模擬式教學的定義

　　模擬式教學依照Alessi和Trollip(2001)的定義是將一些現象或活動呈現給學生，並且讓學生能夠與模擬環境互動，以進行學習活動。De Jong 和Van Joolingen(1998)認為，模擬式教學是一個程式，它呈現系統性的模型或過程，模擬可以廣泛地分為兩類：第一、融入概念模型的模擬；第二、融入操作模型的模擬；概念模型中，會有學科的概念或原則，像是經濟學、物理學的原理；操作模型中，會有認知或非認知的操作流程，像是雷達操作。模擬的過程中，可以改變輸入的變數達成學習上所需的輸出狀態，或是檢驗輸出狀態，模擬可以融入非線性結構的學習目標，學習目標一旦達成，學生仍然能持續和反覆調整變數(O'Neil, Wainess, & Baker, 2005)。

　　學生透過模擬，在操作、調整參數的過程中，可以檢驗自己習得的概念，在模擬工具給予回饋時產生認知衝突，或是有機會在傳統實驗或高風險實驗以外的情況，了解實驗的變數與過程，學生也能夠反覆運用視覺化模型反思自己的概念，並且不斷詢問「如果…」與「為什麼」這類的問題(Chen, Hong, Sung, & Chang, 2011; Huppert, Yaakobi, & Lazarowitz, 1998; Thomas, & Neilson, 1995)。

二、模擬式教學之實施

　　模擬式教學能夠提升學生互動與操作的機會，這些互動與操作能夠增進學生的學習成效(Chen, Hong, Sung, & Chang, 2011)。Chen, Hong, Sung和Chang(2011)將模擬式教學的模式(如圖二)，分成三個面向：概念學習(concept learning)、模擬操作(simulative manipulation)、概念澄清(concept clarification)，學生透過參數的調整學習到概念，在學生有基礎概念後，用模擬工具達到更詳細的概念澄清，在這過程中，學生可以經歷反思的循環。



圖二、模擬式教學的模型  
(Chen, Hong, Sung, & Chang, 2011)

　　在教學的過程中，模擬式教學能夠幫助學生理解抽象概念，因為一些靜態圖像難以表達動態的抽象概念，像是在醫學模擬訓練中，動態的模擬更能夠使學生理解藥物對血壓造成的影響(Krishnan, Keloth, & Ubedulla, 2017)，視覺化的模擬工具也能夠幫助學生了解電腦硬體之間的運作的抽象概念，讓學生清晰地理解電腦科學領域中的抽象定義(Garay et al., 2017)，在模擬環境中，學生能夠反覆操作，更能促進精熟學習，比起講述式的課程，知識與技巧的留存會更加持久，模擬給予學生適當的訓練，根據學生自己的學習情況調整學習步調，促使學生能自主學習，讓他們在面對與模擬情境類似的情況時，更有信心完成任務(Hošková-Mayerová & Rosická, 2015)。

　　在科學教育方面，模擬環境的即時輸入與輸出，讓學生判斷系統上各種元素間的關係，這些關係是根據清楚界定的假設，與傳統實驗不同的是即時的呈現，能夠讓學生更容易得知假設與實驗結果間的關係(Mintz, 1993)，在模擬工具上，對學生要求更多探索，會讓學生處理模擬工具的過程更好，而且也會有更深層的學習效果，探索的過程也會讓學生的理解更好，學生與模擬工具互動程度也與他們轉換到真實實驗的學習成就有顯著相關(Homer & Plass, 2014)，教學內容如果牽涉到複雜的計算，但教學目的是期待學生了解一個整體性的概念，模擬工具可以協助一些計算過程，減輕學生的認知負荷，讓他們在與模擬工具互動時，更加專注於整體的概念運作(Moyer-Packenham et al., 2019)。

　　根據上述優點，因為本研究的教學主題為人工智慧中的類神經網路，認為類神經網路的運算架構屬於抽象概念，且學生需要反覆的演算類神經網路的架構進而熟悉此概念，所以使用模擬式教學，在類神經網路視覺化的模擬環境中，讓學生自主操作得到即時的輸出反饋，了解到類神經網路這個抽象概念，而模擬環境也會協助運算類神經網路中的數學概念，使學生更著重於學習類神經網路的整體概念。

第四節 演算法視覺化

　　演算法視覺化是用圖像化的方式呈現演算法或程式動態改變的運行過程，視覺化的主要目的是幫助學生更加理解演算法(Tudoreanu et al., 2002)。在電腦科學教育中，視覺化也時常被討論是否能夠提升教學效率(Diehl, 2007; Tudoreanu et al., 2002; Shaffer et al., 2010)。

　　在過去演算法視覺化的相關研究中，有部分研究說明這類教學對於學生的學習成效沒有顯著地幫助(Gurka & Citrin, 1996; Hundhausen & Douglas, 2000; Jarc, Feldman, & Heller, 2000) 在演算法視覺化的教材中，若給學生建立自己的視覺化作品會使他們分心，而且視覺化的美觀程度還有是否使用電腦作為媒介對於教學成效沒有顯著差異(Hundhausen & Douglas, 2000)。與演算法視覺化教學系統有互動機會的學生，相較於沒有機會互動的學生而言，可能在某些類型的問題上表現良好，但有互動機會的學生需要花費較多時間在這些教材上，通盤來說仍然表現較差(Jarc et al., 2000)。有的研究則明確地指出這類教學是確實有幫助到學生學習的(Shaffer et al., 2010; Lawrence, Badre, & Stasko, 1994; Hundhausen, Douglas, & Stasko, 2002)，對於演算法視覺化的長期研究也理解出使這類教材有效的因素(Shaffer et al., 2010)，像是給予學生控制演算法的執行步驟，就能夠幫助學生拆解演算法的邏輯，進而更深入地理解演算法(Saraiya et al., 2004)，或是透過比喻的方式，將演算法的執行過程比喻成日常的生活經驗，並透過動畫呈現，能夠幫助學生理解，也讓學生能夠長期記憶演算法的概念(Hansen, Narayanan, & Schrimpsher, 2000)。

　　依照上述演算法視覺化的研究成果，本研究在設計模擬式教學輔助工具時，會讓學生能夠操作與觀察類神經網路執行過程中的每個步驟，也會使用較日常生活的例子說明類神經網路的運作原理與應用方式。

參考文獻：

教育部（2018），十二年國民基本教育課程綱要 國民中學暨普通型高級中等學校 科技領域，教育部。

黃仁暐、涂益郎（2019），和AI做朋友-相知篇，教育部。

Simulation

Algorithm Visualization

AI

Abu-Naser, S. S. (2008). Developing visualization tool for teaching AI searching algorithms.

Alessi, S. M., & Trollip, S. R. (2001). Multimedia for learning: Methods and development. Allyn & Bacon.

Barella, A., Valero, S., & Carrascosa, C. (2008). JGOMAS: New approach to AI teaching. IEEE Transactions on education, 52(2), 228-235.

Bellstrom, P., Thoren, C. (2009). Learning how to program through visualization: A pilot study on the bubble sort algorithm. 2009 Second International Conference on the Applications of Digital Information and Web Technologies (Icadiwt 2009), 90–94. doi:10.1109/icadiwt.2009.5273943

Burgsteiner, H., Kandlhofer, M., & Steinbauer, G. (2016, March). Irobot: Teaching the basics of artificial intelligence in high schools. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 30, No. 1).

Chen, Y. L., Hong, Y. R., Sung, Y. T., & Chang, K. E. (2011). Efficacy of simulation-based learning of electronics using visualization and manipulation. *Journal of Educational Technology & Society*, *14*(2), 269-277.

Colaso, V., Kamal, A., Saraiya, P., North, C., McCrickard, S., & Shaffer, C. (2002). Learning and retention in data structures: A

comparison of visualization, text, and combined methods. Paper presented at the Proceedings of ED-MEDIA 2002, June 24-29,

Denver, Colorado, USA.

Cuéllar, M. P., & Pegalajar, M. C. (2014). Design and implementation of intelligent systems with LEGO Mindstorms for undergraduate computer engineers. Computer Applications in Engineering Education, 22(1), 153-166.

De Jong, T., & Van Joolingen, W. R. (1998). Scientific discovery learning with computer simulations of conceptual domains. *Review of educational research*, *68*(2), 179-201.

Diehl, S. (2007). Software visualization: visualizing the structure, behaviour, and evolution of software. Springer Science & Business Media.

Esteves, M., Fonseca, B., Morgado, L., & Martins, P. (2011). Improving teaching and learning of computer programming through the use of the Second Life virtual world. British Journal of Educational Technology, 42(4), 624-637.

Estevez, J., Garate, G., Guede, J. M., & Grana, M. (2019). Using Scratch to Teach Undergraduate Students' Skills on Artificial Intelligence. arXiv preprint arXiv:1904.00296.

Fernandes, M. A. (2016). Problem‐based learning applied to the artificial intelligence course. Computer Applications in Engineering Education, 24(3), 388-399.

Garay, G. R., Tchernykh, A., Drozdov, A. Y., Garichev, S. N., Nesmachnow, S., & Torres-Martinez, M. (2017). Visualization of VHDL-based simulations as a pedagogical tool for supporting computer science education. *Journal of Computational Science*.

Grivokostopoulou, F., Perikos, I., & Hatzilygeroudis, I. (2014, December). Using semantic web technologies in a web based system for personalized learning AI course. In 2014 IEEE Sixth International Conference on Technology for Education (pp. 257-260). IEEE.

Gurka, J. S., & Citrin, W. (1996, September). Testing effectiveness of algorithm animation. In Proceedings 1996 IEEE Symposium on Visual Languages (pp. 182-189). IEEE.

Hansen, S. R., Narayanan, N. H., & Schrimpsher, D. (2000). Helping learners visualize and comprehend algorithms. Interactive Multimedia Electronic Journal of Computer-Enhanced Learning, 2(1), 10.

Homer, B. D., & Plass, J. L. (2014). Level of interactivity and executive functions as predictors of learning in computer-based chemistry simulations. *Computers in Human Behavior*, *36*, 365-375.

Hošková-Mayerová, Š., & Rosická, Z. (2015). E-learning pros and cons: active learning culture?. *Procedia-Social and Behavioral Sciences*, *191*, 958-962.

Hundhausen, C., & Douglas, S. (2000, September). Using visualizations to learn algorithms: should students construct their own, or view an expert's?. In Proceeding 2000 IEEE International Symposium on Visual Languages (pp. 21-28). IEEE.

Hundhausen, C. D., Douglas, S. A., & Stasko, J. T. (2002). A meta-study of algorithm visualization effectiveness. Journal of Visual Languages & Computing, 13(3), 259-290.

Huppert, J., Yaakobi, J., & Lazarowitz, R. (1998). Learning microbiology with computer simulations: Students’ academic achievement by method and gender. *Research in Science & Technological Education*, *16*(2), 231-245.

Jacob, S. R., & Warschauer, M. (2018). Computational thinking and literacy. Journal of Computer Science Integration, 1(1).

Jarc, D. J., Feldman, M. B., & Heller, R. S. (2000). Assessing the benefits of interactive prediction using web-based algorithm animation courseware. ACM SIGCSE Bulletin, 32(1), 377-381.

Jensen, D., Self, B., Rhymer, D., Wood, J., & Bowe, M. (2002). A rocky journey toward effective assessment of visualization modules for learning enhancement in Engineering Mechanics. Journal of Educational Technology & Society, 5(3), 150-162.

Jonassen, D. H., & Strobel, J. (2006). Modeling for meaningful learning. In Engaged learning with emerging technologies (pp. 1-27). Springer, Dordrecht.

Kandlhofer, M., Steinbauer, G., Hirschmugl-Gaisch, S., & Huber, P. (2016, October). Artificial intelligence and computer science in education: From kindergarten to university. In 2016 IEEE Frontiers in Education Conference (FIE) (pp. 1-9). IEEE.

Kochlán, M., & Hodon, M. (2014, September). Open hardware modular educational robotic platform—Yrobot. In 2014 23rd International Conference on Robotics in Alpe-Adria-Danube Region (RAAD) (pp. 1-6). IEEE.

Korhonen, A., & Malmi, L. (2000). Algorithm Simulation with Automatic Assessment. Paper presented at the 5th Annual ACM

SIGCSE/SIGCUE Conference on Innovation and Technology in Computer Science Education (ITiCSE 2000). Helsinki, Finland.

Krishnan, D. G., Keloth, A. V., & Ubedulla, S. (2017). Pros and cons of simulation in medical education: A review. *Education*, *5*, 7.

Kumar, A. N. (2004). Three years of using robots in an artificial intelligence course: lessons learned. Journal on Educational Resources in Computing (JERIC), 4(3), 2-es.

Lawrence, A. W., Badre, A. M., & Stasko, J. T. (1994, October). Empirically evaluating the use of animations to teach algorithms. In Proceedings of 1994 IEEE Symposium on Visual Languages (pp. 48-54). IEEE.

Marković, M., Kostić Kovačević, I., Nikolić, O., & Nikolić, B. (2015). INSOS—educational system for teaching intelligent systems. Computer Applications in Engineering Education, 23(2), 268-276.

McNally, M., Naps, T., Furcy, D., Grissom, S., & Trefftz, C. (2007). Supporting the rapid development of pedagogically effective algorithm visualizations. Journal of Computing Sciences in Colleges, 23(1), 80-90.

Mintz, R. (1993). Computerized simulation as an inquiry tool. Scool Science and Mathematics, 93(2), 76-80.

Moyer-Packenham, P. S., Lommatsch, C. W., Litster, K., Ashby, J., Bullock, E. K., Roxburgh, A. L., ... & Clarke-Midura, J. (2019). How design features in digital math games support learning and mathematics connections. *Computers in Human Behavior*, *91*, 316-332.

Naps, T. L., Rößling, G., Almstrum, V., Dann, W., Fleischer, R., Hundhausen, C., et al. (2003). Exploring the role of visualization and engagement in computer science education. ACM SIGCSE Bulletin, 35(2), 131-152.

O'Neil, H. F., Wainess, R., & Baker, E. L. (2005). Classification of learning outcomes: Evidence from the computer games literature. *The Cirriculum Journal*, *16*(4), 455-474.

Pedro, F., Subosa, M., Rivas, A., & Valverde, P. (2019). Artificial intelligence in education: Challenges and opportunities for sustainable development.

Prensky, M. (2002). The Motivation of Gameplay or, the REAL 21th century learning revolution. URL http://www. marcprensky. com/writing/Prensky, 2010-1.

Russell, S., & Norvig, P. (2002). Artificial intelligence: a modern approach.

Saraiya, P., Shaffer, C. A., McCrickard, D. S., & North, C. (2004, March). Effective features of algorithm visualizations. In Proceedings of the 35th SIGCSE technical symposium on Computer Science Education (pp. 382-386).

Seehorn, D., Carey, S., Fuschetto, B., Lee, I., Moix, D., O'Grady-Cunniff, D., ... & Verno, A. (2011). CSTA K--12 Computer Science Standards: Revised 2011. ACM.

Selby, C., & Woollard, J. (2013). Computational thinking: the developing definition.

Shaffer, C. A., Cooper, M. L., Alon, A. J. D., Akbar, M., Stewart, M., Ponce, S., & Edwards, S. H. (2010). Algorithm visualization: The state of the field. ACM Transactions on Computing Education (TOCE), 10(3), 1-22.

Simoňák, S. (2016, January). Algorithm visualizations as a way of increasing the quality in computer science education. In 2016 IEEE 14th international symposium on applied machine intelligence and informatics (SAMI) (pp. 153-157). IEEE.

Sklar, E., Eguchi, A., & Johnson, J. (2002, June). RoboCupJunior: learning with educational robotics. In Robot Soccer World Cup (pp. 238-253). Springer, Berlin, Heidelberg.

Stone, P., Brooks, R., Brynjolfsson, E., Calo, R., Etzioni, O., Hager, G., ... & Teller, A. (2016). Artificial intelligence and life in 2030: the one hundred year study on artificial intelligence.

Thomas, R., & Neilson, I. (1995). Harnessing simulations in the service of education: The interact simulation environment. *Computers & Education*, *25*(1-2), 21-29.

Tudoreanu, M. E., Wu, R., Hamilton-Taylor, A., & Kraemer, E. (2002, September). Empirical evidence that algorithm animation promotes understanding of distributed algorithms. In Proceedings IEEE 2002 Symposia on Human Centric Computing Languages and Environments (pp. 236-243). IEEE.

Tversky, B., Morrison, J. B., & Betrancourt, M. (2002). Animation: can it facilitate?. International journal of human-computer studies, 57(4), 247-262.

Végh, L., & Stoffová, V. (2017). Algorithm animations for teaching and learning the main ideas of basic sortings. Informatics in Education, 16(1), 121-140.

Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), 33-35.

Yoon, D. M., & Kim, K. J. (2015). Challenges and opportunities in game artificial intelligence education using Angry Birds. IEEE Access, 3, 793-804.