

## 中文摘要

工程圖的數位化與自動標註是智慧製造的關鍵步驟，其中標註資訊的精確提取對於自動化製程設計至關重要。為了提取標註資訊，需要定位與辨識工程圖上的向量文字。此外，為了理解工程圖含意，需要將多個字串組合後同時進行解讀。這些工作均需要耗費大量人力與時間，雖然，已有研究針對工程圖自動化標註提出了一套系統，但是，印刷電路板佈局圖 (Printed Circuit Board Layout, P.C.B. Layout) 等工程圖中的文字常被元件或線段遮擋，而且，文字組合方式因格式而異，使得自動化辨識仍面臨挑戰。現有方法中，點陣圖光學字元辨識 (Optical Character Recognition, OCR) 方法將圖像轉為點陣圖後進行辨識，但是，點陣化會將線段投影至同一平面，使重疊線段混合，造成遮擋處的文字難以辨識。使用向量資料來進行辨識，其含有線段的連接與幾何資訊，能避免線段合併的問題，但是，現有向量物件偵測方法依賴物件的線段組成方式固定，對於視覺上相似但線段組成不同的文字難以處理。因此，本研究提出基於線段連接關係的點雲 (PointCloud) 偵測方法，將線段取樣為點並記錄其連接關係與斜率特徵，透過調整特徵聚合策略，優先聚合相連線段取樣而成的點，使遮擋處的文字能獲取足夠的連接結構特徵，克服點陣圖方法與現有向量物件偵測方法的限制。此外，現有系統採用基於規則的方法進行文字組合，利用空間關係配對字串，但是，遇到新格式需要修改規則，缺乏適應性。本研究提出兩階段自動配對方法，結合語意特徵與相對位置、尺寸等空間特徵，自動學習組合規則，能適應不同格式的工程圖。最後，本研究分兩階段進行驗證。針對文字辨識，與商業光學字元辨識 (OCR) 系統，包括，谷歌光學字元辨識 (Google OCR)、(Character Region Awareness For Text detection, CRAFT) 等比較，在遮擋處與整體準確率均優於現有方法，證明本研究方法能有效處理遮擋文字。針對文字組合，與基於規則的方法比較，能達到同等的組合準確率，證實自動配對機制的可行性。

# 第 1 章 緒論

在現代製造業中，工程圖不僅是產品設計的藍圖，更是串聯設計、生產與品質控制的關鍵文件，架起工程師、製造人員與品管人員之間的溝通橋梁。印刷電路板佈局圖 (P.C.B. Layout) 是電子製造領域中常見的工程圖形式，通常以可攜式文件格式 (Portable Document Format, PDF) 等向量格式儲存與傳遞，其以幾何元素（如線段、曲線、圓弧等）的數學描述來表示圖形，但缺乏工程語意的結構化資訊。典型的印刷電路板佈局圖 (P.C.B. Layout) 包含元件編號、針腳 (Pin) 位置與編號、尺寸標註等資訊，這些資訊在自動化生產流程中用於指導設備進行元件定位與組裝，在品質檢驗階段則作為驗證產品是否符合設計規範的依據。在實務上，代工廠接收客戶提供的工程文件後，需要從印刷電路板佈局圖 (P.C.B. Layout) 中提取相關標註資訊以建立生產與檢驗參數。傳統作法多仰賴人工判讀與手動輸入，然而當圖檔數量增加或圖面內容較為複雜時，此流程不僅耗時，也容易因人為疏失導致資訊錯誤，進而影響後續製程。因此，開發自動化的圖面資訊擷取方法，有助於提升資料處理效率與準確性。由於印刷電路板佈局圖 (P.C.B. Layout) 上的標註資訊主要以文字形式呈現，而這些文字在向量格式中通常被轉換為由線段與曲線組成的幾何形狀，無法直接提取文字內容，本研究的首要目標為擷取工程圖上的文字內容，作為後續資訊解析與應用的基礎。

過去已有研究提出工程圖自動化標註系統，但是，針對擷取工程圖中的文字，採用的是在點陣圖上的光學字元辨識 (OCR)，將向量圖轉換為點陣圖後進行辨識。然而，點陣化過程會將所有文字與其他元素投影至同一平面並合併為像素，導致原本在向量格式中獨立存在的重疊物件無法區分。如圖 1.1 所示，印刷電路板佈局圖 (P.C.B. Layout) 上的文字常面臨遮擋問題：元件編號常被針腳 (Pin) 遮擋，尺寸標註則常被箭頭線段遮擋，使得被遮擋的文字難以辨識。為了避免點陣化問題，因此，本論文採用直接處理向量格式的資料，利用向量資料中保留的幾何元素資訊與空間關係。儘管向量資料保留了完整的幾何資訊，但是，文字辨識仍面臨技術挑戰。向量格式中的文字由多個基本幾何元素（主要為線段與曲線）組合而成，而現有向量物件偵測方法通常假設同一類物件由相同類型與數量的幾何元素以一致的拓撲結構組成。如圖 1.2 所示，同一字元可能有多種組成方式：字元 0 的曲線可能由貝茲曲線 (Bézier curve) 或短直線構成。當視覺外觀相似但是線段組成不同時，現有向量物件偵測方法會作為兩種不同的物件。當視覺外觀相似但幾何元素組成不同時，現有向量物件偵測方法會將其視為不同類別的物件。即使嘗

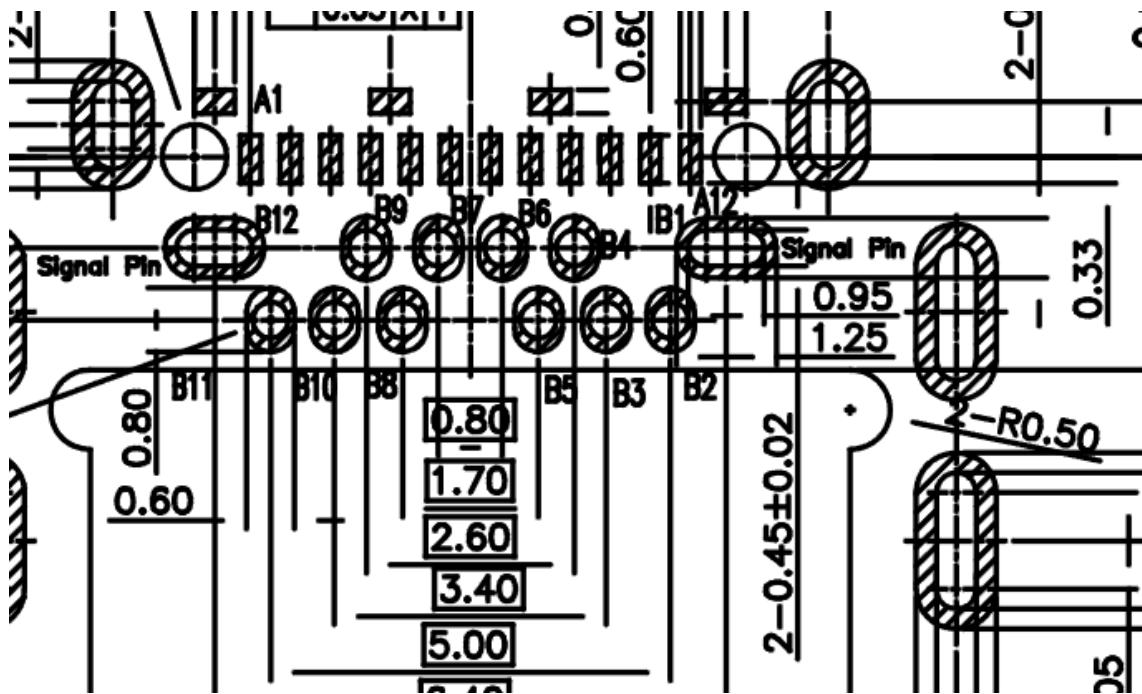


圖 1.1: 工程圖中的向量文字資訊容易遭到遮擋。元件編號 (如圖中的 B4,B5,B7,B9,A12 等)遭到針腳或線段遮擋。尺寸資訊 (如 0.80, 2-R0.50)遭到線段或針腳遮擋。

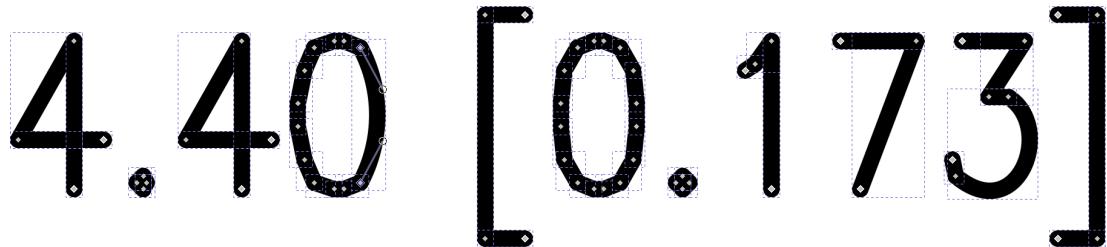


圖 1.2: 視覺上相似的文字，可能有多種組成方式。圖片中，左側的 0 的組成包含了貝茲曲線 ( Bézier curve )，右側的 0 由純直線組成。

試列舉所有可能的組合方式，仍可能遇到未預期的組合，導致偵測失敗。因此，本論文提出將向量資料轉換為帶有幾何資訊的點雲表示法，在保留向量格式中幾何元素類型與空間關係等結構資訊的同時，將不同組成方式的字元統一為基於採樣點的表示形式，使得物件偵測模型能夠專注於字元的整體形狀特徵，而非依賴特定的幾何元素組合模式，從而提升對結構變異的適應能力。然而，工程圖上的標註資訊通常由多個字元或符號組合而成，需要進一步將這些元素正確配對與解讀，才能提取完整的工程語意。以尺寸標註的公差資訊為例，其完整表示往往包含主數值、正負公差值、數量標記等多個元素，這些元素可能採用不同的排列方式與字體大小：部分採用上下標結構，部分則以平行方

式排列，僅依賴空間關係無法準確判斷其組合邏輯。如圖 1.3 所示，這些標註皆用於表達尺寸的主數值與公差範圍，但是，採用了不同的排列方式：圖 1.3(A) 將正負公差平行排列於主數值右側，圖 1.3(B) 將正負公差以上下標形式標註，而圖 1.3(C) 如同 1.3(B)，也以上下標形式標註，卻未以負號標註負公差值。現有系統採用基於規則的配對方法，根據空間關係與格式特徵設計規則。然而，遇到新的標註格式時需要重新設計規則，增加系統開發與維護成本。為了自動化處理此類標註資訊的組合問題，本研究提出兩階段深度學習配對方法，分別透過學習字元間的空間關係與語意關聯，將字元組合成完整標註，避免針對每種格式設計專門規則。圖 ??fig: 究的整體流程。

$0.81 \pm 0.05$	$0.8_{-0}^{+0.1}$	$2.7_{0}^{+0.1}$
-----------------	-------------------	------------------

圖 1.3: 公差值存在不同的排列方式。(A) 使用對稱公差符號表示  $\pm$  (B) 將正負公差以上下標形式標註 (C) 以正號標註正公差，卻未以負號標註負公差值

## 1.1 問題定義

本研究聚焦於向量印刷電路板佈局圖 (P.C.B. Layout) 的字元辨識與標註組合。工程圖包含大量的標註資訊，準確提取這些資訊是實現自動化製程的基礎。本研究將標註資訊提取分為兩個核心任務：向量字元辨識與標註組合。向量字元辨識需要從工程圖的線段中偵測，並且，識別單個字元與符號。由於工程圖以向量格式儲存，字元由參數化線段組成，這帶來了兩個主要挑戰。首先，字元常被其他元件或線段遮擋，遮擋處的線段在空間上與其他線段重疊。為了正確辨識遮擋處的字元，方法需要利用線段之間的連接關係與斜率資訊，這些幾何特性在向量格式中得以保留，是準確判斷字元結構的基礎。其次，同一字元在不同工程圖中可能有多種線段組成方式。例如，字元 E 可能由兩個方框組成，也可能由方框與橫線組成；字元 0 的曲線可能由貝茲曲線 (Bézier curve) 曲線或短直線構成。這些組成方式在視覺上相似，但線段的數量、連接方式與幾何參數各異。為了穩定辨識不同來源的工程圖，方法需要能夠處理這種多樣性，不依賴固定的線段組成模式。除了字元辨識，標註組合同樣面臨挑戰。工程圖中的標註資訊通常需要多個字元組合才能表達完整語意。例如，尺寸標註包含主數值、公差與數量標記，這些元素的排

列方式與字體大小因格式而異：有些採用上下標結構，有些則平行排列，僅依賴空間距離無法準確判斷歸屬關係。為了實現自動組合，方法需要結合字元的語意特徵（如類別）與空間特徵（如位置、尺寸）來判斷組合關係，並且能夠適應不同格式的標註結構。當遇到新的標註格式時，方法應能透過學習自動適應，降低系統開發與維護成本。本研究的目標是：輸入向量工程圖  $V$ ，經本研究方法  $M$  處理後，輸出標註資訊  $A$ 。其中  $A$  包含所有標註的位置座標、文字內容與組合關係，能夠直接應用於後續的自動化製程與品質控制。

## 1.2 主要貢獻

綜上所述，本論文主要貢獻統整如下

1. 提出基於線段連接關係的點雲偵測方法，改善遮擋文字的辨識：工程圖上的向量文字由參數化線段組成，同一字元的視覺外觀可能對應多種線段組成形式，且文字常與其他元件或線段重疊造成遮擋。現有點陣圖方法在點陣化過程中將所有線段投影至同一平面並轉換為像素，導致重疊線段無法區分，丟失線段的連接關係與斜率資訊，使得遮擋處難以辨識。現有向量物件偵測方法則依賴物件的線段組成方式固定，難以處理視覺相似但組成不同的字元。為了克服這些限制，本研究將線段取樣為點雲表示，記錄每個線段的端點連接關係與斜率資訊。透過調整點雲深度學習模型的分組策略，在特徵聚合階段優先聚合相連線段取樣而成的點，使模型能在遮擋處獲取足夠的連接結構特徵與斜率分布。同時，由於模型學習的是線段的連接模式與斜率特性，而非固定的線段組成方式，因此能夠處理視覺相似但組成不同的字元，提升辨識的穩定性與準確性。
2. 提出兩階段將字元組成標註，以取得標註內容工程圖的標註格式多樣，包含不同的上下標結構、公差表示方式與數量標記等。現有系統採用基於規則的配對方式，需要針對不同格式手動分析並設計規則，當遇到新的標註格式時需要重新設計規則，增加系統開發與維護成本。本研究提出兩階段配對機制：模型透過學習字元的空間與語意特徵，自動將字元依空間關係串連成字串，再依語意邏輯歸納為完整標註。當遇到新的標註格式時，僅需提供訓練資料即可適應，無需重新設計規則，降低系統開發與維護成本，提升方法的通用性。

## 1.3 論文架構

本論文包含五個章節，內容簡述如下：

- **第一章，介紹**本章從電子製造業的自動化需求出發，說明工程圖標註資訊擷取的重要性與技術挑戰，提出向量工程圖標註資訊擷取需求。並延伸出其在現有方法的問題與挑戰，包括字元遮擋、結構變異與格式多樣性，並且，說明本研究採用點雲深度學習與兩階段配對機制的設計理念。
- **第二章，相關研究**本章回顧工程圖標註資訊擷取的相關技術與研究。首先討論現有工程圖自動化系統的架構與限制，接著回顧文字辨識技術，包括點陣圖光學字元辨識 (OCR) 與向量物件偵測方法，分析兩者在處理工程圖上的限制。最後討論文字組合方法，說明本研究方法與現有技術的差異。
- **第三章，方法總覽**本章說明本研究方法的整體架構與核心設計決策。首先總覽工程圖自動化標註的應用需求與技術挑戰，接著說明選擇點雲深度學習模型的原因、調整點雲模型以處理字元遮擋與結構變異的設計理念，以及兩階段配對方法適應標註格式多樣性的設計邏輯，為理解第四章的實作細節建立基礎。
- **第四章，向量文字辨識與組合方法**本章詳細說明三個核心模組的設計與實作細節。首先介紹字元檢測模組的線段取樣策略、分組策略調整與斜率特徵編碼；接著介紹字元識別模組的類別與角度預測方法及損失函數設計；最後介紹兩階段配對模組的架構，說明如何透過分離空間與語意的學習適應標註格式的多樣性。
- **第五章，實驗結果與討論**介紹實驗資料集與評估指標，說明訓練與驗證資料的處理方式。透過消融測試驗證各項設計的有效性，並將本研究方法與商業光學字元辨識 (OCR) 系統及基於規則的組合方法比較，證實本研究方法的準確率與實用性。
- **第六章，結論與未來工作**總結論文並討論限制與未來工作。

## 第 2 章 相關研究

工程圖的數位化與自動化標註是智慧製造的關鍵技術，長期以來受到學術界與工業界的關注。在此領域中已有大量研究，涵蓋光學字元辨識 (OCR)、物件偵測、文件版面分析等，在各類工程圖的辨識與資訊提取上取得了顯著成果。為了減少印刷電路板佈局圖 (P.C.B. Layout) 等工程圖標註資訊提取的人工介入，本研究著重於提升向量文字辨識與字串組合的準確率與適應性。在本章節中，本論文將介紹現有工程圖自動化系統的技術路線，分析點陣圖光學字元辨識 (OCR) 方法與向量物件偵測方法的原理與限制，並探討文字組合方法在不同標註格式下的適應性問題，分析其於印刷電路板佈局圖 (P.C.B. Layout) 上的缺陷，同時，說明本研究克服缺陷的方式。

### 2.1 工程圖自動化系統

工程圖的自動化標註是智慧製造的關鍵環節，其目標是從工程圖中提取標註資訊，包含尺寸、公差、數量等製程參數，以支援後續的自動化設計與品質控制。為了實現自動化標註，系統需要能夠辨識工程圖中的文字，並且，將多個字元組合成完整的標註資訊。托羅等人 (Toro et al.) 等人 [1,2] 提出針對工程圖的文字辨識與標註系統。該系統採用點陣圖光學字元辨識 (OCR) 技術與基於規則的配對方法，處理流程包含三個階段：首先將向量格式的工程圖轉換為點陣圖，接著使用光學字元辨識 (OCR) 技術檢測與辨識文字，最後透過預先定義的空間規則將字串組合成標註資訊。在後續研究中 [2]，該系統進一步整合視覺語言模型處理特定類型的標註資訊，提升系統的辨識能力。然而，針對印刷電路板佈局圖 (P.C.B. Layout) 等包含密集線路與元件的工程圖，文字常與線段或元件重疊，造成遮擋問題。點陣圖方法的主要限制在於點陣化過程會將所有重疊的線段合併成單一像素區域，丟失原始的幾何資訊與線段連接關係，導致遮擋處的文字難以辨識。此外，基於規則的配對方法需要針對特定格式設計規則，當遇到新的標註格式時需要重新分析與修改規則，缺乏適應性。為了克服這些限制，本研究選擇在向量資料上進行辨識，保留幾何資訊與線段連接資訊，以偵測出與元件重疊的文字。同時，設計二階段的深度學習模型，使其能夠自主學習組合字元的規則，避免額外的人工設計需求。

## 2.2 文字辨識技術

### 2.2.1 點陣圖光學字元辨識 (OCR) 方法

光學字元辨識 (Optical Character Recognition, OCR) 是一種將圖像中的文字轉換為可編輯文字的技術。傳統光學字元辨識 (OCR) 系統分為文字偵測與文字辨識兩個階段：偵測階段定位文字區域，辨識階段將文字區域轉換為字元序列。早期，張等人 (Zhang et al.) [3] 依賴人工設計的特徵與規則，例如，邊緣檢測、連通區域分析等，對於字體變化與背景干擾較為敏感。近年來，深度學習方法大幅提升了光學字元辨識 (OCR) 的準確率與魯棒性。白等人 (Baek et al.) 等人提出 [4]，採用卷積神經網路預測字元的區域與字元之間的連接關係，能夠處理任意形狀與排列的文字。此方法透過預測字元級別的熱圖 (Heatmap) 與連接熱圖 (Affinity map)，將文字檢測轉換為像素級別的分類問題，在場景文字與文件圖像中表現優異。商業光學字元辨識 (OCR) 系統，如谷歌光學字元辨識 (Google OCR)，結合大規模訓練資料與模型優化，在一般文字辨識任務中達到較高的準確率，已廣泛應用於文件數位化、票據識別等領域。然而，這些方法均需要將輸入資料轉換為點陣圖。在點陣化過程中，向量格式的線段會被轉換為像素，當多條線段在空間上重疊時，所有線段會合併成單一像素區域。如圖 1.1 所示，當文字被其他線段或元件遮擋時，點陣化後的圖像中遮擋處的線段會與文字線段合併，丟失線段的連接關係與斜率等幾何資訊，導致光學字元辨識 (OCR) 系統難以正確分割與辨識字元。此外，點陣化會引入解析度依賴性：過低的解析度會丟失細節特徵，過高的解析度則增加計算成本與記憶體需求。為了避免點陣化帶來的資訊損失，需要直接處理向量格式的資料。向量資料保留了線段的端點座標、連接關係與幾何參數，這些資訊是判斷字元結構的重要依據。接下來介紹向量物件偵測方法的原理與限制。

### 2.2.2 向量物件偵測方法

向量格式的工程圖由多種參數化線段組成，包含不同的曲線與直線，不同於點陣圖的規則網格結構，向量資料的表示形式更為抽象。為了從向量資料中檢測物件，方法需要能夠理解線段之間的幾何關係與拓撲結構。現有研究主要分為兩類：基於節點關係的方法與基於點雲表示的方法。江等人 Jiang et al.) [5] 提出向量圖形辨識方法，採用基於節點關係的策略處理向量資料。此方法將所有線段統一表示為三次貝茲曲線 (Cubic Bezier Curve)，每條曲線由起點、終點與兩個控制點定義。接著，方法將每個端點視為節點，透過分析節點之間的連接關係與線段的幾何特徵，建立圖結構 (Graph)。具體而言，方

法計算節點之間的歐氏距離、曲線的曲率（由控制點位置決定）與線段的方向資訊，作為圖的邊特徵。透過圖神經網路 (Graph Neural Network, GNN) 聚合節點特徵，模型能夠預測哪些線段屬於同一物件。此方法在向量圖表與向量圖示辨識等任務中表現優異，證明了基於節點關係的策略在向量物件偵測中的有效性。然而，此方法依賴物件的線段組成方式保持一致。如圖??fig: 一字元的視覺外觀可能對應多種線段組成方式：字元 E 可能由兩個方框組成，或由方框與橫線組成；字元 0 的曲線可能由一條三次貝茲曲線 (Cubic Bezier Curve) 或多條短直線構成。當線段的數量、連接方式與幾何參數改變時，節點之間的距離與曲率特徵會產生顯著差異，導致模型將視覺相似的字元判定為不同物件。此外，此方法從節點層級分析線段關係，聚合的是局部的節點特徵，難以捕捉整體的連接模式與拓撲結構。對於印刷電路板佈局圖 (P.C.B. Layout) 中組成方式多樣的文字，辨識的穩定性與準確率仍有不足。另一類方法將線段轉換為點雲表示，利用點雲深度學習模型處理。劉等人 Liu et al.) 提出簡易的點雲 [6] 方法，將向量線段取樣為點，並為每個點附加三原色 (Red Green Blue, RGB) 顏色特徵與線段的方向資訊，透過進階點網路 (PointNet++) [7] 提取點的局部與全局特徵。透過多尺度分組 (Multi-Scale Grouping) 聚合鄰近點的特徵，能夠捕捉物件的幾何結構。具體而言，以不同的半徑範圍查詢鄰近點，透過共享權重的多層感知機 (Multi-Layer Perceptron, MLP) 提取局部特徵，並透過最大池化 (Max Pooling) 聚合為中心點的特徵。此方法將向量資料統一表示為點的空間分布，避免了對線段組成方式的依賴，在符號辨識任務中表現良好。然而，此方法未充分考慮線段的連接關係與遮擋問題。在遮擋場景下，遮擋處的點在空間上與其他物件的點混雜。由於模型無法區分哪些點屬於相連線段、哪些點屬於遮擋物件，導致聚合的特徵包含來自不同物件的資訊。在印刷電路板佈局圖 (P.C.B. Layout) 中，文字的線段密度不均且常與其他元件或線段重疊，當遮擋發生時，模型難以在遮擋處獲取足夠的字元結構特徵，導致誤抓或漏抓。此外，雖然該方法為點附加三原色 (RGB) 顏色特徵，但工程圖中的文字與元件通常採用相同或相似的顏色，顏色特徵對於區分遮擋物件的幫助有限。同時，該方法未充分利用線段的斜率與連接模式，這些幾何特性是判斷字元結構的重要依據，特別是在辨識由多條線段組成的複雜字元時。為了克服現有方法的限制，本研究需要在保留點雲表示優勢的同時，引入線段的連接關係與斜率資訊。透過調整特徵聚合策略，使模型能在遮擋處優先獲取相連線段的特徵，同時利用斜率分布判斷字元結構，從而提升辨識的準確率與穩定性。具體設計將在第三章與第四章詳述。

## 2.3 文字組合方法

工程圖的標註資訊通常由多個字串組合而成，例如尺寸標註包含主數值、公差與數量標記，這些元素的排列方式與字體大小因格式而異。僅依賴文字辨識結果無法直接理解標註的完整語意，需要進一步將字串組合成有意義的標註單元。此任務在文件理解領域稱為版面分析 (Layout Analysis) 或文字分組 (Text Grouping)。現有工程圖自動化系統採用基於規則的配對方法 [1] [2]，透過分析字串的空間關係與格式特徵設計規則。具體而言，此類方法計算字串之間的距離、相對位置（如上下、左右）與字體大小比例等特徵，根據預定義的規則判斷組合關係。例如，當檢測到一個較大的數值字串與兩個較小的字串分別位於其上下方時，系統判定為主數值與正負公差的組合；當檢測到數量標記（如”X4”）位於數值字串附近時，將其配對為數量資訊。此類方法透過明確的規則定義組合邏輯，在特定格式的工程圖中表現穩定。然而，基於規則的方法面臨適應性問題。如圖??fig: 同來源的印刷電路板佈局圖 (P.C.B. Layout) 採用不同的標註格式：公差可能採用上下標結構（例如主數值”1.50”，正公差”+0.1”位於上方，負公差”-0.0”位於下方），也可能採用對稱表示（例如” $1.5\pm0.05$ ”），或平行排列（例如”0.8 +0.1 -0”）。這些格式在空間排列、字體大小比例與元素間距上存在顯著差異，單一規則難以涵蓋所有情況。當遇到新的標註格式時，現有規則可能失效，導致錯誤的組合結果。為了適應新格式，需要重新分析標註的空間特徵與組合邏輯，手動設計或修改規則，增加系統開發與維護成本。為了提升方法的通用性與適應性，需要能夠自動學習組合規則的方法。此類方法應能夠結合字串的語意特徵（如字元類別、數值或符號）與空間特徵（如相對位置、尺寸比例）進行配對，透過訓練資料學習不同格式的組合模式。當遇到新的標註格式時，僅需提供訓練資料即可適應，無需重新設計規則。具體方法將在第三章與第四章詳述。

## 第3章 方法總覽

過去的點陣圖光學字元辨識方法受限於點陣化過程會將重疊線段合併，丟失線段的連接關係與幾何資訊，導致遮擋處的文字難以辨識。現有的向量物件偵測方法，例如基於節點關係的方法，依賴物件的線段組成方式保持一致，對於視覺相似但組成不同的字元難以處理。為了實現印刷電路板 (Printed Circuit Board, PCB) 工程圖的自動化標註，本研究設立了向量文字辨識與字串組合兩大目標。為了達成以上目標，本研究提出基於線段連接關係的點雲偵測方法與兩階段自動配對方法。

向量文字辨識方法基於點雲表示與進階點網路 (PointNet++) 架構，藉由本研究提出的改進取樣與分組策略，在遮擋處優先聚合相連線段取樣而成的點，使模型能獲取足夠的連接結構特徵與斜率分布。並且，針對向量字元由線段組成的特性，透過基於筆畫的字元辨識方法，將字元表示為筆畫序列，結合變換器 (Transformer) 架構進行分類與角度預測，克服組成方式多樣性的問題。在本章節中，將探討本論文所提出的向量文字辨識與字串組合方法，說明本研究選擇點雲表示與進階點網路 (PointNet++) 的原因，並且，針對標準進階點網路 (PointNet++) 在遮擋場景下的問題，說明改進的設計理念。最後，為了達成字串自動組合的目標，介紹本研究所提出的兩階段配對方法。整個系統流程，如圖 3.4。

### 3.1 向量文字辨識的技術基礎

#### 3.1.1 點雲表示

工程圖的向量文字由參數化線段組成，同一字元可能對應多種線段組成方式。如圖 1.3 所示，字元 E 可能由兩個方框組成，或由方框與橫線組成；字元 O 的曲線可能由一條三次貝茲曲線 (Cubic Bezier Curve) 或多條短直線構成。這些組成方式在視覺上相似，但線段的數量、連接方式與幾何參數各異。為了處理這種多樣性，需要一種能夠統一表示不同組成方式的資料格式。點雲表示將線段離散化為空間中的點集合。具體而言，對於向量圖中的每條線段，以固定間隔對線段進行取樣，得到一組空間座標點。每個點記錄其二維座標 (x,y) 與附加特徵。透過點雲表示，不同組成方式的字元被統一為點的空間分布，模型學習的是點的排列模式與幾何特徵，而非固定的線段組合方式。這種表示方式

具有以下優勢：首先，點雲表示對線段的組成方式不敏感。無論字元由幾條線段構成，取樣後都形成相似的空間點分布，模型能夠識別點的幾何排列而非線段的拓撲結構。其次，點雲格式保留了線段的空間資訊。每個點的座標反映了線段在空間中的位置，透過點的分布可以重建字元的整體形狀。此外，點雲表示允許為每個點附加額外特徵，例如點所屬線段的斜率、線段的連接關係等，這些特徵是判斷字元結構的重要依據。

### 3.1.2 進階點網路 ( PointNet++ ) 架構

進階點網路 ( PointNet++ ) 是一種分層點雲特徵提取網路，設計用於處理無序點集。相較於傳統的規則網格資料 ( 如影像 )，點雲資料缺乏固定的拓撲結構，因此需要能夠適應任意排列的特徵提取方法。進階點網路 ( PointNet++ ) 透過分層結構，逐層提取局部與全局特徵，適合處理複雜的幾何結構。如圖 3.1 所示，進階點網路 ( PointNet++ ) 的核心是 Set Abstraction 層，每一層透過三個步驟聚合局部特徵：

1. **取樣 ( Sampling )**：使用最遠點取樣 ( Farthest Point Sampling, FPS ) 選擇一部分點作為中心點。最遠點取樣 ( FPS ) 從點集中迭代選擇與已選點距離最遠的點，確保中心點在空間中均勻分布，能夠代表整體的幾何結構。
2. **分組 ( Grouping )**：對每個中心點，以固定半徑查詢鄰近點，形成局部區域。常見的分組方法包括球形查詢 ( Ball Query ) 與 K 最近鄰 ( K-Nearest Neighbors, KNN ) ( K-Nearest Neighbors, K 最近鄰 ( KNN ) )。球形查詢在半徑內選取所有點，確保局部區域的大小一致；K 最近鄰 ( KNN ) 選取距離最近的 K 個點，適應點密度變化。
3. **特徵提取 ( Feature Learning )**：對每個局部區域，透過共享權重的多層感知機 ( Multi-Layer Perceptron, MLP ) 提取特徵，並透過最大池化 ( Max Pooling ) 聚合為中心點的特徵。這種方式使模型能學習局部的幾何模式。進階點網路 ( PointNet++ ) 提出多尺度分組 ( Multi-Scale Grouping, 多尺度分組 ( Multi-Scale Grouping, MSG ) ) 策略，以不同半徑同時查詢鄰近點。例如，以半徑  $r_1, r_2, r_3$  分別形成三個局部區域，提取不同尺度的特徵後串接，使模型能同時捕捉細節與整體結構。透過多層 Set Abstraction，進階點網路 ( PointNet++ ) 逐步減少點的數量並增加特徵維度，最終提取全局特徵用於分類或分割任務。進階點網路 ( PointNet++ ) 的分層結構能夠同時捕捉局部細節與全局結構，適合處理工程圖中組成複雜的字元。此外，進階點網路 ( PointNet++ ) 對點的排列順序不敏感，透

過最大池化聚合特徵，無論點的輸入順序如何，輸出結果保持一致，符合向量線段無固定順序的特性。

### 3.1.3 分析

在本研究的觀察下，認為點雲表示結合進階點網路 ( PointNet++ ) 架構具有以下優勢。首先，點雲表示將不同組成方式的字元統一為點的空間分布，避免了對固定線段組合的依賴。其次，進階點網路 ( PointNet++ ) 的分層結構能夠捕捉多尺度的幾何特徵，適應不同尺寸與複雜度的字元。此外，進階點網路 ( PointNet++ ) 對點的排列順序不敏感，符合向量線段無固定順序的特性，能夠穩定地從點雲中提取字元特徵。相較於基於節點關係的方法 ( 如 YOLaT )，點雲表示不需要將線段統一為特定參數形式 ( 如三次貝茲曲線 ( Cubic Bezier Curve ) )，避免了參數轉換帶來的資訊損失與不一致性。因此，本研究選擇點雲表示與進階點網路 ( PointNet++ ) 作為向量文字辨識的基礎架構。然而，標準的進階點網路 ( PointNet++ ) 在遮擋場景下存在問題。如圖 3.2 所示，當文字被其他線段遮擋時，遮擋處的點在空間上與遮擋物件的點混雜。標準的多尺度分組策略依據歐氏距離查詢鄰近點，無法區分哪些點屬於相連線段、哪些點屬於遮擋物件。因此，聚合的特徵包含來自不同物件的資訊，導致模型在遮擋處難以獲取正確的字元結構特徵。此外，標準的進階點網路 ( PointNet++ ) 僅利用點的空間座標進行分組，未考慮線段的連接關係與斜率資訊。這些幾何特性是區分子元結構的重要依據，特別是在辨識由多條線段組成的複雜字元時。此外，現有的取樣與分組方法也存在限制。在取樣方面，雖然最遠點取樣 ( FPS ) 能夠在空間中均勻分布中心點，但在線段密度不均的場景下，部分線段組合可能未被充分取樣，導致模型缺乏足夠的訓練樣本。有研究提出泊松盤取樣 ( Poisson Disk Sampling, PDS )，要求每個取樣點必須間隔一定距離，在點數量多時計算效率優於最遠點取樣 ( FPS )，但對於本研究的資料未能解決線段組合覆蓋問題。也有研究提出自適應取樣 ( Adaptive Sampling )，透過偏移消除離群點，但在無雜訊時效果劣於 FPS。在分組方面，K 最近鄰 ( KNN ) 方法選取 K 個最近的點，符合空間局部性，但在重疊處無法為不同線段挑選較相關的特徵點；在稀疏空間會取到過遠的點；當半徑內存在多個物件時，會偏重線段最多的物件。基於分區的方法 ( partition-based ) 直接對區域切塊進行取樣與分組，能提升效率，但未解決遮擋場景下的特徵混雜問題。為了解決這些問題，本研究在取樣階段追蹤所有線段組合的取樣數量，確保每種組合都被充分取樣。在分組階段，本研究引入線段的連接關係，透過優先聚合相連線段取樣而成的點，使模型能在遮擋處獲取足夠的字元結構特徵。此外，本研究在點特徵中編碼線段的斜率資訊，使模型能利用線段的方向分布判斷字元結構。具體設計將在 3.2 詳述。

## 3.2 設計

### 3.2.1 改進的取樣與分組策略

#### 改進的取樣策略

標準的最遠點取樣 (FPS) 在取樣時僅考慮點的空間分布，未考慮線段組合的多樣性。在工程圖中，不同來源的字元可能有多種線段組成方式，如果某些組合在訓練資料中出現頻率較低，標準最遠點取樣 (FPS) 可能無法充分取樣這些組合，導致模型對罕見組合的辨識能力不足。本研究提出改進的取樣策略，基於最遠點取樣 (FPS) 但在取樣過程中追蹤所有線段組合的取樣數量。具體而言，在取樣初期，方法依據標準最遠點取樣 (FPS) 選擇空間上分散的點。隨著取樣進行，當接近目標取樣數量時，方法檢查各線段組合的取樣狀況，若發現某些組合的取樣數不足，則優先從這些組合中選擇新的取樣點。透過這種策略，確保每種線段組合都能被取樣到至少一次，提升模型對不同組成方式的適應性。



#### 改進的分組策略

標準的進階點網路 (PointNet++) 採用多尺度分組 (MSG) 策略，以多個半徑進行球形查詢或 K 最近鄰 (KNN)，聚合不同尺度的特徵。然而，在遮擋場景下，這些方法僅依據歐氏距離選點，無法區分相連線段與遮擋物件的點。本研究提出改進的分組策略，採用多重分組形式，同時利用兩種分組方法進行選點，並在取出特徵後將兩種特徵串接。第一種方法延續標準的 K 最近鄰 (KNN) 與球形查詢，以距離進行選點，捕捉空間局部特徵。第二種方法引入線段連接關係，事前計算線段的端點連接資訊，在分組時優先選取相連線段取樣而成的點。具體而言，對於每個中心點，方法首先識別該點所屬的線段，接著查詢與該線段相連的其他線段，並從這些相連線段取樣的點中選取特徵點。透過這種策略，即使在遮擋處多條線段重疊，模型仍能優先聚合屬於同一字元的相連線段點，獲取正確的連接結構特徵。此外，本研究在點特徵中編碼線段的斜率資訊。對於每條線段，計算其單位方向向量 ( $\cos\theta, \sin\theta$ )，並將此向量作為該線段上所有取樣點的附加特徵。透過編碼斜率資訊，模型能利用線段的方向分布判斷字元結構，例如區分豎線與橫線、識別字元的筆畫走向。結合空間座標、連接關係與斜率特徵，模型能在遮擋處有效區分不同物件，提升辨識準確率。

### 3.2.2 基於筆畫的字元辨識

經過點雲偵測模組後，系統得到一組候選字元區域，每個區域包含若干條線段。為了辨識字元的類別與旋轉角度，需要進一步提取字元的語意特徵。向量字元由線段組成，這些線段在視覺上對應手寫字元的筆劃受到手寫字元辨識與手繪圖形辨識研究的啓發，本研究將字元表示為筆畫序列，每條線段視為一個筆畫單元。這種表示方式具有以下優勢：首先，筆畫表示能夠捕捉字元的拓撲結構，不同筆畫之間的連接關係反映了字元的形狀特徵。其次，筆畫序列對線段的數量與組成方式不敏感，模型學習的是筆畫的排列模式而非固定的線段數量。本研究採用變換器 (Transformer) 架構處理筆畫序列。具體而言，每條線段（筆畫）編碼為一個 token，包含線段的空間座標、長度、斜率等幾何特徵。將所有筆畫 token 輸入變換器 (Transformer) 編碼器，透過自注意力機制 (self-attention) 捕捉筆畫之間的關聯性。變換器 (Transformer) 的自注意力機制能夠建模長距離依賴關係，適合處理筆畫數量不固定的字元。除了字元類別預測，本研究還引入角度預測模組。工程圖中的文字可能以不同角度排列，準確預測字元的旋轉角度對於後續的字串組合至關重要。角度預測模組與類別預測共享變換器 (Transformer) 編碼器的特徵，透過額外的輸出層預測字元的旋轉角度。透過聯合訓練類別與角度預測，模型能同時學習字元的語意與幾何特性。



### 3.2.3 兩階段自動配對方法

工程圖的標註資訊通常由多個字串組合而成，例如尺寸標註包含主數值、公差與數量標記，這些元素的排列方式與字體大小因格式而異。現有系統採用基於規則的配對方法，需要針對特定格式設計規則，缺乏適應性。本研究提出兩階段自動配對方法，結合語意特徵與空間特徵進行配對。第一階段將字元組合成字串，第二階段將字串組合成字串組。

- **第一階段：字元組合成字串：**第一階段依據字元的空間排列將連續字元組合成基礎字串。具體而言，方法計算相鄰字元之間的距離與對齊方向，當多個字元在空間上連續排列且距離小於閾值時，將其組合為一個字串。此階段主要利用空間特徵，適合處理水平或垂直排列的連續字元。
- **第二階段：字串組合成字串組：**第二階段處理更複雜的組合關係，例如主數值與上下標公差的配對。由於這些元素的排列方式與字體大小因格式而異，單純依賴空間關係難以準確判斷。本研究提出基於學習的配對方法，結合字串的語意特徵與空間特徵。語意特徵包含字串的字元類別（數字、符號、字母）與字元數量等資訊。

訊；空間特徵包含字串之間的相對位置、尺寸比例與對齊方向等。方法為每對候選字串提取特徵向量，透過神經網路預測配對的可能性。具體而言，對於每個字串，首先提取其語意特徵與空間特徵。接著，對於每對候選字串，計算它們的相對位置向量與尺寸比例，並將兩個字串的特徵與相對特徵串接，形成配對特徵向量。將配對特徵輸入分類器，預測該對字串是否應組合為字串組。透過在訓練資料上學習，模型能夠自動捕捉不同格式的組合模式，當遇到新的標註格式時，模型能利用學到的語意與空間特徵進行推理，無需重新設計規則。

此外，為了提升辨識準確率，在字串組合後，系統對字串進行二次辨識修正。由於部分字元在視覺上相似（例如數字 1 與字母 I、數字 0 與字母 O、負號-與數字 1），單字元辨識可能產生錯誤。透過字串層級的語境資訊，系統能修正這些混淆，例如當字串為”R10”時，即使單字元辨識將”1”誤判為”I”，透過語境分析能修正為正確的數字。



## 第 4 章 研究方法

為了實現印刷電路板 (PCB) 工程圖的自動化標註，第一章定義了兩個核心任務：向量文字辨識與字串組合。第三章說明了選擇點雲表示與進階點網路 ( PointNet++ ) 架構的原因，並提出改進的取樣與分組策略、基於筆畫的字元辨識方法，以及兩階段自動配對方法的設計理念。然而，現有向量物件偵測方法在遮擋場景下會混雜不同物件的特徵，導致偵測準確率下降。此外，現有系統採用基於規則的配對方法，當遇到新的標註格式時需要重新設計規則，缺乏適應性。為了克服這些限制，本研究提出基於線段連接關係的點雲偵測方法與兩階段自動配對方法。本研究將向量文字辨識任務進一步拆分為字元偵測與字元辨識兩個子任務。字元偵測模組負責從點雲中定位字元區域，透過改進的採樣與分組策略，在遮擋處優先聚合相連線段的點，使模型能獲取正確的連接結構特徵。字元辨識模組負責預測字元的類別與旋轉角度，透過基於筆畫的表示與變換器 (Transformer) 架構，處理線段數量不固定的字元。字串組合任務同樣拆分為字元配對與字串配對兩個階段。第一階段將空間上連續排列的字元組合成基礎字串；第二階段結合語意特徵與空間特徵，將基礎字串組合成完整的標註資訊。透過學習配對規則，模型能自動適應不同格式的標註結構，降低系統開發與維護成本。在本章節中，將詳細說明三個核心模組的設計與實作。首先，4.1節介紹字元偵測模組，包含線段預處理、改進的最遠點取樣 (FPS) 採樣策略、基於連接關係的分組策略、斜率特徵編碼，以及基於連接組件的後處理方法。接著，4.2節介紹字元識別模組，說明基於筆畫的字元表示、變換器 (Transformer) 架構設計，以及字元類別與角度的聯合預測方法。最後，4.3節介紹兩階段配對模組，詳述字元配對的空間特徵計算、字串配對的語意與空間特徵提取，以及配對網路的架構與訓練策略。整個系統流程如圖 4.1 所示。

### 4.1 字元偵測模組

字元偵測模組負責從工程圖的向量線段中偵測出文字區域。本模組基於改進的進階點網路 ( PointNet++ ) 架構，透過四個階段完成偵測任務：預處理階段將向量線段轉換為點雲表示並建立連接關係；採樣階段確保每個連接組件都有足夠的代表點；分組階段在遮擋場景下優先聚合相連線段的點；輸出階段基於連接組件進行決策合併。圖 4.1 展示字元偵測模組的完整流程。

相較於第三章的設計理念說明，本節將詳細闡述每個階段的技術實作、參數設定與設計理由。

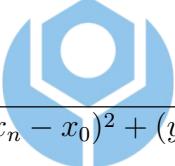
### 4.1.1 預處理

預處理階段將工程圖的向量資料轉換為適合進階點網路 (PointNet++) 處理的點雲格式，並建立線段的連接關係與斜率特徵。此階段包含四個步驟：曲線轉直線、連接分組、斜率計算與線段取樣。

#### 曲線轉直線

工程圖的向量資料包含多種類型的曲線與線段，包括直線 (Line)、二次貝茲曲線 (Quadratic Bezier)、三次貝茲曲線 (Cubic Bezier) 與圓弧 (Arc)。為了統一處理，本研究將所有曲線離散化為直線段序列。

對於類型為直線 ( $type = 0$ ) 的線段，直接保留其起點與終點。對於曲線，本研究採用等間隔採樣策略。具體而言，對於一條曲線  $C$ ，計算其起點  $P_0$  與終點  $P_n$  之間的歐氏距離  $d$ ：


$$d = \sqrt{(x_n - x_0)^2 + (y_n - y_0)^2} \quad (4-1)$$

根據預設的採樣長度  $L_{\text{sample}} = 1.0$ ，計算需要放置的線段數量：

$$N_{\text{seg}} = \lceil d/L_{\text{sample}} \rceil \quad (4-2)$$

接著，在曲線參數空間  $t \in [0, 1]$  上等間隔採樣  $N_{\text{seg}} + 1$  個點：

$$t_i = \frac{i}{N_{\text{seg}}}, \quad i \in \{0, 1, \dots, N_{\text{seg}}\} \quad (4-3)$$

對於二次貝茲曲線，由控制點  $P_0, P_1, P_2$  定義，曲線上的點計算為：

$$P(t) = (1 - t)^2 P_0 + 2(1 - t)t P_1 + t^2 P_2 \quad (4-4)$$

對於三次貝茲曲線，由控制點  $P_0, P_1, P_2, P_3$  定義：

$$P(t) = (1 - t)^3 P_0 + 3(1 - t)^2 t P_1 + 3(1 - t)t^2 P_2 + t^3 P_3 \quad (4-5)$$

將相鄰採樣點連接，得到一組直線段。透過這種離散化方式，所有曲線都被轉換為直線段的序列表示。在本研究的實驗中，採樣長度  $L_{\text{sample}} = 1.0$  能在保持曲線形狀特徵的同時，避免產生過多的線段數量。圖 4.2 展示不同採樣長度對字元「O」的離散化結果。

### 連接分組

工程圖中的字元由多條線段組成，這些線段透過端點連接形成字元的拓撲結構。為了保留線段之間的連接關係，本研究採用聯集查找 (Union-Find) 演算法將連接的線段分組。

兩條線段  $l_i$  與  $l_j$  被視為連接，當且僅當它們至少共享一個端點。由於向量資料的浮點數精度問題，直接比較端點座標可能因為微小誤差而判定為不連接。為了提升魯棒性，本研究將端點座標離散化到整數網格。

具體而言，對於端點座標  $(x, y)$ ，定義座標鍵值函數：



$$\text{key}(x, y) = (\lfloor x \times 10000 \rfloor, \lfloor y \times 10000 \rfloor) \quad (4-6)$$

其中  $\lfloor \cdot \rfloor$  表示四捨五入取整。當兩個端點的座標鍵值相同時，判定為同一個端點。縮放因子 10000 的選擇基於工程圖的典型座標範圍，能夠在容忍微小誤差的同時，避免將不相連的端點誤判為連接。

連接分組採用聯集查找 (Union-Find) 資料結構實現。初始時，每條線段自成一組。對於工程圖中的所有線段，依次檢查其端點：首先計算線段  $l_i$  的起點與終點的座標鍵值，接著查詢是否有其他線段的端點具有相同的座標鍵值。如果存在，使用 Union 操作將這些線段合併到同一組。經過所有線段的處理後，每個連接組件對應一個聯集查找 (Union-Find) 中的根節點。為每條線段分配其所屬組件的 ID，記為  $g_i \in \{1, 2, \dots, G\}$ ，其中  $G$  為連接組件的總數。

連接分組的結果將用於後續的採樣與分組策略。圖 4.3 展示字元「E」的連接分組結果，可以看到組成字元的所有線段被正確地分配到同一個連接組件。

## 斜率計算

線段的斜率是判斷字元結構的重要特徵，特別是在辨識具有明確筆畫方向的字元（如「E」、「F」的橫線與豎線）時。然而，直接使用角度  $\theta \in [0, 2\pi)$  表示斜率會遇到周期性問題：例如  $\theta = 0^\circ$  與  $\theta = 180^\circ$  在視覺上表示相同方向（水平線），但在數值上相差  $180^\circ$ 。

為了去除這種  $180^\circ$  的方向模糊性，本研究採用雙角度轉換 (Double-Angle Transformation)。具體而言，對於一條線段  $l_i$ ，起點為  $(x_1, y_1)$ ，終點為  $(x_2, y_2)$ ，定義方向向量：

$$A = y_1 - y_2, \quad B = x_2 - x_1 \quad (4-7)$$

$$L = \sqrt{A^2 + B^2} \quad (4-8)$$

歸一化後得到單位方向向量：



$$\hat{A} = A/L, \quad \hat{B} = B/L \quad (4-9)$$

計算原始角度：

$$\theta = \text{atan2}(\hat{B}, \hat{A}) \quad (4-10)$$

其中  $\theta \in [-\pi, \pi]$ 。為了去除方向性，將角度限制到  $[-\pi/2, \pi/2]$ ：

$$\theta' = \begin{cases} \theta & \text{if } \theta \in [-\pi/2, \pi/2] \\ \theta - \pi & \text{if } \theta > \pi/2 \\ \theta + \pi & \text{if } \theta < -\pi/2 \end{cases} \quad (4-11)$$

接著，應用雙角度轉換：

$$\theta_2 = 2\theta' \quad (4-12)$$

最終的斜率特徵為：

$$A' = \cos(\theta_2), \quad B' = \sin(\theta_2), \quad C' = 0 \quad (4-13)$$

透過這種編碼方式， $\theta' = 0^\circ$  與  $\theta' = 180^\circ$  (都表示水平線) 會得到相同的  $(A', B')$  值。具體而言，當  $\theta' = 0^\circ$  時， $\theta_2 = 0^\circ$ ， $(A', B') = (1, 0)$ ；當  $\theta'$  經過限制後等效於  $0^\circ$  時，亦得到相同結果。這種編碼方式消除了方向模糊性，使模型能專注於學習線段的傾斜角度而非方向。圖 4.4 展示不同角度線段的斜率特徵編碼範例。

### 線段取樣

為了將線段轉換為點雲格式，本研究對每條線段進行取樣，生成一組空間點。取樣策略需要平衡兩個目標：保留線段的幾何特徵（端點與整體形狀）與控制點的數量（避免過度採樣導致計算負擔）。

本研究採用基於長度的自適應取樣策略。對於一條線段  $l_i$ ，長度為  $L_i$ ，計算需要取樣的點數。最小取樣間隔設定為  $L_{\min} = 0.25$  或  $0.5$  (可配置)，每條線段的最大取樣數設定為  $N_{\max} = 5$ 。取樣點數的計算公式為：

$$N_i = \min \left( \left\lceil \frac{L_i}{L_{\min}} \right\rceil, N_{\max} \right) \quad (4-14)$$

這種策略確保：第一，長線段得到更多取樣點，保留其形狀特徵；第二，短線段至少有 2 個點（起點與終點）；第三，不會因為極長的線段產生過多取樣點。

對於線段  $l_i$ ，在起點  $P_1 = (x_1, y_1)$  與終點  $P_2 = (x_2, y_2)$  之間進行線性插值：

$$P_s = P_1 + \frac{s}{N_i - 1} (P_2 - P_1), \quad s \in \{0, 1, \dots, N_i - 1\} \quad (4-15)$$

每個取樣點  $P_s$  附加以下特徵形成完整的點特徵向量：

$$f_s = [x_s, y_s, 0, A', B', C', L_i, g_i] \quad (4-16)$$

其中  $(x_s, y_s, 0)$  為點的三維座標 ( $z = 0$  因為工程圖是平面的)， $(A', B', C')$  為所屬

線段的斜率特徵 (由前述雙角度轉換計算)， $L_i$  為所屬線段的長度， $g_i$  為所屬線段的連接組件 ID。因此，每個點的特徵維度為 8。

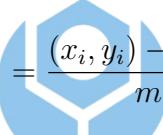
在輸入進階點網路 (PointNet++) 之前，需要對點雲進行歸一化處理。具體而言，對於一個包含  $N$  個點的點雲  $\mathcal{P} = \{P_1, P_2, \dots, P_N\}$ ，計算其中心點：

$$\bar{P} = \frac{1}{N} \sum_{i=1}^N P_i \quad (4-17)$$

計算最大距離：

$$m = \max_i \sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2} \quad (4-18)$$

歸一化座標：



$$(x'_i, y'_i) = \frac{(x_i, y_i) - (\bar{x}, \bar{y})}{m} \quad (4-19)$$

這種歸一化方式將所有點約束到以原點為中心、半徑約為 1 的圓內，使不同尺寸的字元具有相似的座標尺度。斜率特徵  $(A', B', C')$  已經是歸一化的方向向量，無需額外處理。

經過預處理後，工程圖被轉換為一組點雲  $\mathcal{P}$ ，每個點包含座標、斜率、長度與組件 ID 等特徵，為進階點網路 (PointNet++) 的處理做好準備。圖 4.5 展示預處理的完整流程與中間結果。

### 4.1.2 改進的採樣策略

第三章 3.2.1 節指出，標準的最遠點採樣 (Farthest Point Sampling, FPS) 僅考慮點的空間分布，未考慮線段組件的代表性。在工程圖中，字元可能包含多個連接組件 (例如字元「i」包含點與豎線兩個組件)，如果某些組件的點數較少，標準 FPS 可能無法充分取樣這些組件，導致模型在特徵聚合時缺乏足夠的組件資訊。

為了解決此問題，本研究提出線段組感知最遠點取樣 (Line Group-Aware FPS)，在採樣過程中追蹤每個連接組件的採樣數量，確保每個組件都有最少的代表點。

圖 4.1: 字元偵測模組的完整流程。從向量線段輸入，經過預處理、進階點網路 ( PointNet++ ) 特徵提取，最終輸出文字區域的檢測結果。

圖 4.2: 不同採樣長度對曲線離散化的影響。(a) 原始貝茲曲線；(b)  $L_{\text{sample}} = 0.5$ ；(c)  $L_{\text{sample}} = 1.0$ ；(d)  $L_{\text{sample}} = 2.0$ 。過小的採樣長度產生過多線段，過大的採樣長度丟失曲線細節。



圖 4.3: 聯集查找 (Union-Find) 連接分組結果。(a) 字元「E」的線段分布；(b) 連接分組後，相同顏色表示屬於同一連接組件。可以看到字元的所有筆畫被正確分配到同一組件。

圖 4.4: 不同角度線段的斜率特徵編碼範例。展示  $0^\circ$ 、 $45^\circ$ 、 $90^\circ$ 、 $135^\circ$  與  $180^\circ$  角度線段經過雙角度轉換後的  $(A', B')$  值，可以看到  $0^\circ$  與  $180^\circ$  得到相同的編碼。

圖 4.5: 預處理的完整流程。(a) 原始向量線段；(b) 曲線離散化；(c) 連接分組 ( 相同顏色表示同一組件 )；(d) 線段取樣為點雲；(e) 歸一化後的點雲表示。

## 線段組感知最遠點取樣 (Line Group-Aware FPS)

設輸入點雲  $\mathcal{P} = \{p_1, p_2, \dots, p_N\}$ ，每個點  $p_i$  屬於連接組件  $g_i \in \{1, 2, \dots, G\}$ ，目標採樣數量為  $N_{\text{sample}}$ 。線段組感知最遠點取樣 (Line Group-Aware FPS) 的核心思想是在標準 FPS 的基礎上，加入連接組件的採樣數量追蹤與約束。

首先，初始化採樣點集合  $\mathcal{S} = \emptyset$ ，距離陣列  $D[i] = \infty, \forall i \in \{1, \dots, N\}$ ，以及每個組件的計數器  $\text{count}[g] = 0, \forall g \in \{1, \dots, G\}$ 。隨機選擇第一個點  $s_1$  加入採樣集合，並更新其所屬組件的計數。接著，計算所有點到  $s_1$  的距離，更新距離陣列。

根據目標採樣數  $N_{\text{sample}}$  與組件總數  $G$  的關係，設定每個組件的採樣約束。當  $N_{\text{sample}} \geq 3G$  時，採樣配額充足，設定每個組件至少採樣 2 個點 ( $\text{min\_samples} = 2$ )，無最大限制 ( $\text{max\_samples} = \infty$ )。當  $G < N_{\text{sample}} < 3G$  時，採樣配額適中，設定每個組件至少採樣 1 個點 ( $\text{min\_samples} = 1$ )，無最大限制。當  $N_{\text{sample}} \leq G$  時，採樣配額稀疏，無最小要求 ( $\text{min\_samples} = 0$ )，但每個組件最多採樣 1 個點 ( $\text{max\_samples} = 1$ )。

在迭代採樣階段，對於每次採樣，首先檢查是否有組件的採樣數未達到最小要求。若存在這樣的組件，將候選點限制為這些組件中的點，從中選擇距離最遠的點加入採樣集合。若所有組件都已達到最小採樣要求，則執行標準 FPS，但排除已達到最大限制的組件，從剩餘候選點中選擇距離最遠的點。每次選擇新的採樣點後，更新該點所屬組件的計數，並重新計算所有點到採樣集合的最小距離。重複此過程直到採樣數量達到  $N_{\text{sample}}$ 。

在本研究的實驗中，進階點網路 (PointNet++) 第一層的目標採樣數為  $N_{\text{sample}} = 512$ ，而工程圖中的字元組件數通常  $G < 10$ ，因此屬於充足採樣的情況，每個組件至少會被採樣 2 個點。這確保了即使是點數較少的組件 (如字元「i」的點部分)，也能在採樣後保留足夠的代表點。

圖 4.6 展示線段組感知最遠點取樣 (Line Group-Aware FPS) 與標準 FPS 在採樣字元「E」時的結果。標準 FPS 僅根據空間距離選點，可能在字元的某些筆畫上過度採樣 (如長橫線)，而在其他筆畫上採樣不足 (如短橫線)。相較之下，線段組感知最遠點取樣 (Line Group-Aware FPS) 確保每個筆畫 (連接組件) 都有足夠的代表點，使模型能獲取完整的字元結構資訊。

線段組感知最遠點取樣 (Line Group-Aware FPS) 能提升字元偵測的召回率，特別是在遮擋處的改善更為顯著。具體的消融測試結果將在第五章呈現。

### 4.1.3 改進的分組策略

第三章 3.2.1 節提出改進的分組策略，在進階點網路 ( PointNet++ ) 的第一層採用基於連接關係的分組，優先聚合相連線段的點。本節詳細說明這種分組策略的實作方式與技術細節。

#### 多重分組架構

標準的進階點網路 ( PointNet++ ) 採用多尺度分組 (Multi-Scale Grouping, MSG)，以多個半徑同時查詢鄰近點，捕捉不同尺度的幾何特徵。本研究在 MSG 的基礎上，引入基於連接關係的分組，形成雙重分組 (Dual Grouping) 架構。

對於進階點網路 ( PointNet++ ) 的第一層 (SA1)，採用三個尺度的多尺度分組。第一尺度使用半徑 0.1 的標準球形查詢，選取 32 個鄰居，透過多層感知機 (MLP) [32, 32, 64] 提取特徵。第二尺度使用較小的半徑 0.05，選取 64 個鄰居，透過 MLP [64, 64, 128] 提取更細緻的局部特徵。第三尺度採用本研究提出的基於連接關係的分組 (同組查詢 (Same-Group Query))，使用半徑 0.1，選取最多 128 個鄰居，透過 MLP [64, 96, 128] 提取組件級別的結構特徵。

這種配置使模型能同時獲取三種不同層次的資訊：大範圍的空間分布 (半徑 0.1，32 鄰居)、精細的局部細節 (半徑 0.05，64 鄰居)、以及連接組件的結構關係 (同組查詢 (Same-Group Query))。三個尺度的特徵最終串接為 320 維的完整特徵 ( $64 + 128 + 128 = 320$ )。

#### 基於連接關係的分組

標準的球形查詢 (Ball Query) 透過歐氏距離判斷鄰近點，無法區分哪些點屬於相連線段、哪些點屬於遮擋物件。在遮擋場景下，這會導致特徵混雜，影響偵測準確率。

本研究提出同組查詢 (Same-Group Query)，在分組時限制只選取與中心點屬於相同連接組件的點。具體而言，對於中心點  $c_i$ ，其所屬連接組件為  $g_i$ ，在半徑  $r$  內查詢滿足以下條件的點：

$$\mathcal{N}(c_i, r, g_i) = \{p_j \mid \|p_j - c_i\| < r \text{ and } g_j = g_i\} \quad (4-20)$$

其中  $g_j$  為點  $p_j$  所屬的連接組件 ID。透過這種限制，即使在遮擋處多條線段的點在空間上混雜，模型仍能優先聚合屬於同一字元的相連線段點，獲取正確的連接結構特徵。

在實作中，本研究修改標準的球形查詢 (Ball Query) 演算法，加入組件 ID 的判斷。首先，對於查詢點  $q$ ，獲取其所屬組件  $g_q$ 。接著，計算點雲中所有點到查詢點的距離。關鍵的修改在於，對於組件 ID 不同於  $g_q$  的點，將其距離設定為無窮大，從而在後續的鄰居選擇中被自動排除。最後，選取距離最小的  $K$  個點作為鄰居，其中距離小於半徑  $r$  且距離不為無窮大的點會被保留。若選取的鄰居數量不足  $K$  個，進行重複採樣補足。

透過將不同組件的點距離設為無窮大，同組查詢 (Same-Group Query) 能有效地排除遮擋物件的點，確保分組的純淨性。圖 4.7 展示在遮擋場景下，同組查詢 (Same-Group Query) 與標準球形查詢 (Ball Query) 的分組差異。

### 多組件追蹤

在實際場景中，採樣點可能橫跨多個連接組件 (例如在組件的交界處)。為了處理這種情況，本研究為每個採樣點維護一個組件列表，記錄其鄰域內出現的連接組件 ID。

具體而言，對於每個中心點  $c_i$ ，在標準球形查詢 (Ball Query) (第一與第二尺度) 中查詢鄰近點時，記錄這些鄰近點所屬的所有不同組件 ID，最多保留  $M_{\max} = 8$  個組件 ID：

$$G_i = \{g_j \mid p_j \in \mathcal{N}(c_i, r)\}, \quad |G_i| \leq M_{\max} \quad (4-21)$$

在同組查詢 (Same-Group Query) (第三尺度) 中，放寬組件判斷條件，當查詢點  $c_k$  的組件列表  $G_k$  與當前點  $p_j$  的組件  $g_j$  有任何重疊時，允許選取：

$$p_j \in \mathcal{N}(c_k, r, \text{any\_overlap}) \iff g_j \in G_k \text{ or } g_k \in G_j \quad (4-22)$$

這種策略允許模型在組件交界處聚合相關的點，同時仍能在大部分區域保持基於連接關係的分組純淨性。

## 後續層的分組

在進階點網路 ( PointNet++ ) 的第二層 (SA2) 與第三層 (SA3)，本研究恢復使用標準的球形查詢 (Ball Query)，不再限制組件 ID。這是因為：第一，第一層已經為每個採樣點提取了基於連接關係的局部特徵；第二，後續層的採樣點數更少 ( $512 \rightarrow 128 \rightarrow 1$ )，需要聚合更大範圍的特徵；第三，在較高的抽象層次，組件級別的連接資訊已經被編碼到特徵中。

第二層 (SA2) 採樣 128 個點，使用兩個尺度的多尺度分組。第一尺度使用半徑 0.1，選取 64 個鄰居，透過 MLP [128, 128, 256] 提取特徵。第二尺度使用半徑 0.2，選取 128 個鄰居，透過 MLP [128, 196, 256] 提取更大範圍的特徵。兩個尺度的特徵串接為 512 維 ( $256 + 256 = 512$ )。

第三層 (SA3) 採用全局池化 (Global Pooling)，將所有點的特徵聚合為單一全局特徵。具體而言，使用 group\_all 策略，將第二層的 128 個點視為一個整體，透過 MLP [256, 512, 1024] 提取全局特徵。輸入特徵維度為 515 ( 512 來自 SA2 + 3 維 XYZ 座標 )，輸出為 1024 維的全局表示。



透過這種分層設計，模型能在第一層獲取細緻的連接結構特徵，在後續層逐步抽象為全局的語意特徵。圖 4.8 展示不同層的特徵提取過程與感受野變化。

### 4.1.4 特徵提取與輸出

經過採樣與分組後，進階點網路 ( PointNet++ ) 對每個局部區域提取特徵。本節說明特徵提取的具體實作與最終的輸出層設計。

#### 局部特徵提取

對於每個中心點  $c_i$  與其鄰域點集  $\mathcal{N}_i = \{p_1, p_2, \dots, p_K\}$ ，特徵提取包含以下步驟。

首先，將鄰域點的座標相對於中心點進行中心化，得到相對座標：

$$\Delta p_j = p_j - c_i, \quad \forall p_j \in \mathcal{N}_i \quad (4-23)$$

這種中心化操作使模型能學習平移不變的局部幾何模式。

接著，對於每個鄰域點  $p_j$ ，將其相對座標與原始特徵串接：

$$f_j = [\Delta x_j, \Delta y_j, 0, A'_j, B'_j, C'_j, L_j] \quad (4-24)$$

其中  $(\Delta x_j, \Delta y_j, 0)$  為相對座標， $(A'_j, B'_j, C'_j)$  為斜率特徵， $L_j$  為線段長度。因此，每個點的特徵維度為 7。

將特徵輸入共享權重的多層感知機 (Multi-Layer Perceptron, MLP)，提取高層次特徵。以第一尺度為例，MLP 配置為  $[32, 32, 64]$ ，即透過三層全連接網路，將 7 維輸入特徵逐步轉換為 32 維、32 維、最終 64 維的高層特徵。每層全連接後接批次歸一化 (Batch Normalization) 與 ReLU 激活函數。

對鄰域內所有點的特徵進行最大池化 (Max Pooling)，聚合為中心點的局部特徵：

$$h_i = \max_{p_j \in \mathcal{N}_i} h_j^{(3)} \quad (4-25)$$

最大池化操作對點的排列順序不敏感，符合點雲的無序性特徵。

最後，將不同尺度的局部特徵串接，形成中心點的完整特徵：

$$f_i^{\text{SA1}} = [h_i^{\text{scale1}}, h_i^{\text{scale2}}, h_i^{\text{scale3}}] \quad (4-26)$$

對於 SA1，特徵維度為  $64 + 128 + 128 = 320$ 。

### 特徵傳播

進階點網路 (PointNet++) 在提取全局特徵後，需要將特徵傳播回原始點雲，以實現逐點分類。本研究採用插值上採樣 (Interpolation Upsampling) 策略。

對於採樣後的點集  $\mathcal{S} = \{s_1, s_2, \dots, s_M\}$  與原始點集  $\mathcal{P} = \{p_1, p_2, \dots, p_N\}$ ，對於每個原始點  $p_i$ ，找到其在採樣點集中的  $K_{\text{nn}} = 3$  個最近鄰  $\{s_{j_1}, s_{j_2}, s_{j_3}\}$ ，透過距離加權插值計算特徵：

$$w_{j_k} = \frac{1}{\|p_i - s_{j_k}\|^2 + \epsilon}, \quad k \in \{1, 2, 3\} \quad (4-27)$$

$$\bar{w}_{j_k} = \frac{w_{j_k}}{\sum_{k=1}^3 w_{j_k}} \quad (4-28)$$

$$f_i^{\text{interp}} = \sum_{k=1}^3 \bar{w}_{j_k} f_{s_{j_k}} \quad (4-29)$$

其中  $\epsilon = 10^{-8}$  為避免除零的微小常數。

插值特徵與該層輸入的原始特徵串接後，透過 MLP 進一步提取特徵。特徵傳播包含三層。第三層 (FP3) 將 SA3 的 1024 維全局特徵與 SA2 的 512 維特徵串接，輸入維度為 1536，透過 MLP [256, 256] 提取特徵。第二層 (FP2) 將 FP3 的 256 維特徵與 SA1 的 320 維特徵串接，輸入維度為 576，透過 MLP [256, 128] 提取特徵。第一層 (FP1) 將 FP2 的 128 維特徵與原始點雲的 3 維 XYZ 座標串接，輸入維度為 131，透過 MLP [128, 128] 提取最終的逐點特徵。

### 輸出層與分類



經過特徵傳播後，每個原始點都獲得 128 維的特徵。最終透過兩層卷積進行逐點分類。第一層卷積將 128 維特徵轉換為 128 維，後接批次歸一化、ReLU 激活與 Dropout (丟棄率 0.5)。第二層卷積將 128 維特徵轉換為  $C$  維的類別邏輯值 (logits)，其中  $C$  為類別數。在本研究中， $C = 2$ ，分別對應「文字」與「非文字」。最終透過 Log-Softmax 得到類別機率：

$$p_i = \text{LogSoftmax}(\text{logits}_i) \quad (4-30)$$

訓練時使用負對數似然損失 (Negative Log-Likelihood Loss)：

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^N \log p_i^{(y_i)} \quad (4-31)$$

其中  $y_i \in \{0, 1\}$  為點  $p_i$  的真實標籤。真實標籤的生成基於預處理階段記錄的線段標註資訊。在訓練資料準備時，標註人員對原始線段進行標註，標記哪些線段屬於文字，並記錄哪些線段屬於同一個字元。基於這些標註，線段上的所有取樣點繼承該線段的標籤。因此，每個點  $p_i$  的真實標籤  $y_i$  直接來自於其所屬線段的標註。

訓練配置方面，批次大小設定為 16 或 32，總訓練週期為 251 個 epoch。採用 Adam 優化器，初始學習率為 0.001，權重衰減 (Weight Decay) 設定為  $1 \times 10^{-4}$ 。學習率調度採用 StepLR 策略，每 20 個 epoch 將學習率乘以 0.5。資料載入使用 24 個工作程序 (Workers) 進行訓練資料的並行處理，測試時設定為 0 以減少資源消耗。

圖 4.9 展示字元偵測模組的完整特徵提取與輸出流程。

#### 4.1.5 基於連接組件的後處理

進階點網路 (PointNet++) 輸出逐點的分類結果，但在實際應用中，需要將這些逐點預測聚合為字元級別的偵測結果。第三章 3.2.1 節提出基於連接組件的後處理策略，本節詳細說明此策略的實作方式。

##### 組件級別的決策聚合

由於字元由多條連接的線段組成，僅依賴單個點的預測結果容易受到雜訊影響。本研究提出組件級別的決策聚合策略，基於連接組件內所有點的預測結果，判斷整個組件是否為文字。

對於連接組件  $g_i$ ，計算其內部所有點的預測結果。由於訓練資料的標註是對原始線段進行的，且記錄了哪些線段屬於同一個字元，因此在推論階段，本研究採用積極策略：當組件內任何一個點被預測為文字時，判定整個組件為文字。這種策略能提升召回率，避免因為部分點預測錯誤而遺漏文字組件。

具體而言，定義組件的文字分數：

$$\text{score}(g_i) = \max_{p_j \in g_i} p_j^{(\text{text})} \quad (4-32)$$

其中  $p_j^{(\text{text})}$  為點  $p_j$  被預測為文字的機率。當  $\text{score}(g_i) > \tau_{\text{conf}}$  時，判定組件  $g_i$  為文字，其中  $\tau_{\text{conf}} = 0.5$  為信心閾值。這等同於檢查組件內是否存在任何一個點的文字機率超過閾值。

這種積極策略的合理性在於：第一，字元的線段組成方式多樣，部分線段可能較短或特徵不明顯，若要求所有點都被正確預測，會過於嚴格；第二，遮擋場景下，部分點的特徵可能受到干擾，但只要關鍵的連接點被正確預測，即可判定為文字；第三，相較

圖 4.6: 線段組感知最遠點取樣 (Line Group-Aware FPS) 與標準 FPS 的採樣結果比較。  
(a) 字元「E」的原始點雲；(b) 標準 FPS 採樣 512 個點，可以看到長橫線被過度採樣，短橫線採樣不足；(c) 線段組感知最遠點取樣 (Line Group-Aware FPS) 採樣 512 個點，每個筆畫都有足夠的代表點，採樣分布更加均勻。

圖 4.7: 在遮擋場景下的分組策略比較。(a) 字元「E」被線段遮擋；(b) 標準球形查詢 (Ball Query) 在遮擋處同時選取字元與遮擋物的點 (紅色圓圈內)，導致特徵混雜；(c) 同組查詢 (Same-Group Query) 僅選取與中心點屬於相同組件的點，成功排除遮擋物的干擾。



圖 4.8: 進階點網路 ( PointNet++ ) 的分層特徵提取過程。(a) 原始點雲；(b) SA1 提取局部連接特徵，感受野較小；(c) SA2 聚合中尺度特徵；(d) SA3 提取全局特徵。顏色深度表示特徵的抽象程度。

圖 4.9: 字元偵測模組的完整特徵提取與輸出流程。從點雲輸入，經過多層 Set Abstraction 提取分層特徵，透過 Feature Propagation 將特徵傳播回原始點雲，最終輸出逐點分類結果。

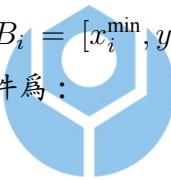
於漏檢文字 (降低召回率)，誤檢少量非文字組件 (降低精確率) 的代價較小，可透過後續的字元辨識模組過濾。

## 相交組件的合併

在實際工程圖中，同一個字元的多個筆畫可能在預處理階段被分配到不同的連接組件 (例如當字元的筆畫未完全連接時)。為了將這些屬於同一字元的組件合併為單一文字區域，本研究提出相交組件合併策略。

首先，篩選出所有被判定為文字的連接組件，形成文字組件集合  $\mathcal{G}_{\text{text}} = \{g_i \mid \text{IsText}(g_i) = \text{True}\}$ 。接著，對於每對文字組件  $(g_i, g_j)$ ，計算其邊界框 (Bounding Box)。邊界框定義為包含組件所有點的最小矩形，由左下角座標  $(x_{\min}, y_{\min})$  與右上角座標  $(x_{\max}, y_{\max})$  確定。

兩個邊界框被判定為相交，當且僅當它們在  $x$  軸與  $y$  軸上的投影均有重疊。具體而言，設組件  $g_i$  的邊界框為  $B_i = [x_i^{\min}, y_i^{\min}, x_i^{\max}, y_i^{\max}]$ ，組件  $g_j$  的邊界框為  $B_j = [x_j^{\min}, y_j^{\min}, x_j^{\max}, y_j^{\max}]$ ，相交條件為：



$$\text{Intersect}(B_i, B_j) = (x_i^{\min} < x_j^{\max}) \wedge (x_i^{\max} > x_j^{\min}) \wedge (y_i^{\min} < y_j^{\max}) \wedge (y_i^{\max} > y_j^{\min}) \quad (4-33)$$

建立相交圖 (Intersection Graph)，其中每個文字組件為節點，相交的組件對之間存在邊。對相交圖進行連通組件分析 (Connected Component Analysis)，找出所有連通的組件集合。對於每個連通組件，計算其所有組件點的聯集，並計算聯集的外接矩形 (Bounding Box) 作為最終的文字區域。

這種合併策略能夠將空間上相近且屬於同一字元的多個組件整合為單一文字區域。圖 4.10 展示相交組件合併的範例。

經過基於連接組件的後處理後，系統輸出一組文字區域，每個區域包含邊界框座標與內部的點雲資料。這些文字區域將輸入字元辨識模組進行進一步的類別與角度預測，此部分將在 4.2 節詳述。

## 4.2 字元辨識模組

字元辨識模組負責預測偵測到的文字區域的類別與旋轉角度。本模組基於變換器 (Transformer) 架構，將字元表示為筆畫序列，每條筆畫由一組連接的線段組成。透過自注意力機制捕捉筆畫之間的關聯性，模型能夠處理線段數量不固定的字元，適應不同組成方式的多樣性。圖 4.11 展示字元辨識模組的完整架構。

字元辨識模組可以獨立訓練，直接從原始線段預處理輸入；也可以在推理階段承接進階點網路 (PointNet++) 的輸出，對偵測到的文字區域進行辨識。本節將詳細說明特徵表示方式、變換器 (Transformer) 架構設計、輸出層配置與多對稱性角度處理策略。

### 4.2.1 筆畫序列表示

第三章 4.2 節提出將字元表示為筆畫序列，每條線段視為一個筆畫單元。本節詳細說明筆畫序列的特徵表示方式。



連接線段的筆畫分組

字元由多條線段組成，這些線段透過端點連接形成字元的拓撲結構。在預處理階段 (4.1.1 節)，已經透過聯集查找 (Union-Find) 演算法識別連接組件，將相連的線段分配到同一組。然而，對於字元辨識而言，連接組件內部的線段排列順序對於理解字元的筆畫結構至關重要。

本研究將連接組件內的線段按照筆畫順序排列。筆畫定義為從起始端點到終止端點的連續線段序列。為了確定筆畫順序，本研究採用端點度數 (Endpoint Degree) 分析與啓發式排序策略。

首先，對於連接組件內的所有線段，計算每個端點的度數，即連接到該端點的線段數量。度數為 1 的端點稱為終端點 (Terminal Point)，表示筆畫的起點或終點；度數為 2 或更高的端點為連接點 (Junction Point)，表示多條線段的交匯處。

筆畫排序策略包含四個步驟。第一步，優先選擇度數為 1 的終端點作為筆畫的起點，確保從字元的邊緣開始。第二步，在同一組件內，按照從上到下 (最大  $y$  座標)、從左到右 (最小  $x$  座標) 的順序選擇起點，提供一致的排序基準。第三步，從起點出發，依次選擇相連的線段，優先選擇與當前線段夾角最小的線段，形成連續的筆畫。第四步，

在添加每條線段時，根據前一條線段的終點，可能需要交換當前線段的起點與終點，確保線段方向的連續性。

透過這種排序策略，連接組件內的線段被轉換為有序的筆畫序列。對於簡單字元（如「I」、「L」），可能只有一條筆畫；對於複雜字元（如「E」、「W」），包含多條筆畫。圖 4.12 展示字元「E」的筆畫排序結果。

### 線段特徵表示

對於排序後的筆畫序列，每條線段  $l_i$  被編碼為 8 維特徵向量。這些特徵包含線段的幾何資訊與在筆畫中的位置資訊。

具體而言，對於線段  $l_i$ ，起點為  $(x_1, y_1)$ ，終點為  $(x_2, y_2)$ ，特徵向量定義為：

$$f_i = [x_1, y_1, x_2, y_2, \cos(\theta_i), \sin(\theta_i), s_i, e_i] \quad (4-34)$$

前四維  $(x_1, y_1, x_2, y_2)$  為線段的起點與終點座標，已經過歸一化處理。歸一化方式為除以字元邊界框的最大邊長  $L_{\max} = \max(W, H)$ ，其中  $W$  與  $H$  分別為字元邊界框的寬度與高度。這種歸一化方式將字元約束到相似的尺度，使模型能學習尺度不變的特徵。

第五與第六維  $(\cos(\theta_i), \sin(\theta_i))$  為線段間角度 (Inter-line Angle) 特徵。此角度表示當前線段與前一條線段之間的夾角，反映了筆畫的轉折程度。對於筆畫中的第一條線段，由於沒有前一條線段，設定  $\cos(\theta_1) = 0, \sin(\theta_1) = 0$ 。對於後續線段，設前一條線段的方向向量為  $\mathbf{v}_1 = (x_2^{(i-1)} - x_1^{(i-1)}, y_2^{(i-1)} - y_1^{(i-1)})$ ，當前線段的方向向量為  $\mathbf{v}_2 = (x_2^{(i)} - x_1^{(i)}, y_2^{(i)} - y_1^{(i)})$ 。線段間角度的正弦值透過二維叉積 (Cross Product) 計算：

$$\sin(\theta_i) = \frac{\mathbf{v}_1 \times \mathbf{v}_2}{\|\mathbf{v}_1\| \|\mathbf{v}_2\|} \quad (4-35)$$

其中二維叉積定義為  $\mathbf{v}_1 \times \mathbf{v}_2 = v_1^x v_2^y - v_1^y v_2^x$ ，結果為標量，其符號表示旋轉方向。餘弦值透過點積 (Dot Product) 計算：

$$\cos(\theta_i) = \frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{\|\mathbf{v}_1\| \|\mathbf{v}_2\|} \quad (4-36)$$

透過同時編碼正弦與餘弦值，模型能完整地表示  $[-\pi, \pi]$  範圍內的角度，避免角度

表示的周期性問題。

第七與第八維  $(s_i, e_i)$  為起始標記 (Start Flag) 與終止標記 (End Flag)，用於指示線段在筆畫中的位置。當線段為筆畫的第一條線段時， $s_i = 1$ ，否則  $s_i = 0$ ；當線段為筆畫的最後一條線段時， $e_i = 1$ ，否則  $e_i = 0$ 。這些標記幫助模型識別筆畫的邊界，理解字元的拓撲結構。

透過這種 8 維特徵表示，每條線段包含了其幾何位置、與前一條線段的關聯性、以及在筆畫中的結構位置，為變換器 (Transformer) 提供了豐富的輸入資訊。

#### 4.2.2 變換器 (Transformer) 架構

第三章 4.2 節提出採用變換器 (Transformer) 架構處理筆畫序列，透過自注意力機制捕捉線段之間的長距離依賴關係。本節詳細說明變換器 (Transformer) 的輸入嵌入、位置編碼、編碼器結構與輸出層設計。



將 8 維的線段特徵向量轉換為變換器 (Transformer) 的輸入嵌入 (Input Embedding)，本研究採用分組投影策略，將不同類型的特徵分別投影後相加。

具體而言，8 維特徵被分為四個部分：座標特徵  $(x_1, y_1, x_2, y_2)$ 、角度特徵  $(\cos(\theta), \sin(\theta))$ 、起始標記  $(s)$  與終止標記  $(e)$ 。每個部分透過獨立的投影層轉換為 128 維的嵌入向量，最終透過逐元素相加 (Element-wise Sum) 融合。

座標投影 (Coordinate Projection) 將 4 維座標特徵透過線性層投影為 128 維。角度投影 (Angle Projection) 將 2 維角度特徵投影為 128 維。起始標記與終止標記為離散值  $s, e \in \{0, 1\}$ ，透過嵌入層 (Embedding Layer) 轉換為 128 綴，嵌入層參數分別為  $E_{\text{start}}, E_{\text{end}} \in \mathbb{R}^{2 \times 128}$ ，根據標記值查詢對應的嵌入向量。

最終，將四個嵌入向量相加，得到線段  $l_i$  的完整嵌入表示：

$$\mathbf{h}_i = \mathbf{h}_{\text{coord}} + \mathbf{h}_{\text{angle}} + \mathbf{h}_{\text{start}} + \mathbf{h}_{\text{end}} \quad (4-37)$$

這種分組投影策略的優勢在於，每種特徵類型能學習專屬的投影權重，避免不同類

型特徵（如連續的座標與離散的標記）直接混合導致的資訊損失。

### 位置編碼

變換器 (Transformer) 架構對輸入序列的順序不敏感，需要透過位置編碼 (Positional Encoding) 為模型提供序列位置資訊。本研究採用正弦位置編碼 (Sinusoidal Positional Encoding)。

對於序列中第  $pos$  個位置，位置編碼為 128 維的向量，其第  $i$  維 ( $i \in \{0, 1, \dots, 127\}$ ) 定義為：

$$PE(pos, 2i) = \sin\left(\frac{pos}{10000^{2i/128}}\right) \quad (4-38)$$

$$PE(pos, 2i + 1) = \cos\left(\frac{pos}{10000^{2i/128}}\right) \quad (4-39)$$

偶數維度使用正弦函數，奇數維度使用餘弦函數。這種編碼方式使模型能學習相對位置關係，因為對於任意固定的偏移  $k$ ， $PE(pos + k)$  可以表示為  $PE(pos)$  的線性函數。

位置編碼與輸入嵌入相加，形成最終的輸入表示：

$$\mathbf{x}_i = \mathbf{h}_i + PE(i) \quad (4-40)$$

本研究設定最大序列長度為 512，即支援最多 512 條線段的字元。在實驗資料中，99% 的字元線段數小於 100，因此此設定能涵蓋絕大多數情況。

### {變換器 (Transformer)}

{變換器 (Transformer) 由 4 層堆疊而成，每層包含多頭自注意力機制 (多頭自注意力 (Multi-Head Self-Attention)) 與前饋網路 (Feed-Forward Network)。隱藏維度設定為 128，採用 8 個注意力頭，每個頭的維度為 16。前饋網路的隱藏層維度為 512。每個子層後使用殘差連接 (Residual Connection) 與層歸一化 (Layer Normalization) 穩定訓練過程，並應用 Dropout (丟棄率 0.1) 防止過擬合。

對於輸入序列  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_L]$ ，其中  $L$  為序列長度，多頭自注意力首先將輸入投影為查詢 (Query)、鍵 (Key)、值 (Value) 矩陣，再分割為 8 個頭並行計算縮放點積注意力 (Scaled Dot-Product Attention)。每個頭計算注意力分數後，將結果串接並透過輸出投影層得到最終結果。前饋網路採用兩層全連接網路，中間使用 ReLU 激活函數，將維度從 128 擴展到 512 再還原為 128。

經過 4 層編碼器後，序列中的每個位置都獲得了融合全局上下文的表示。

### 4.2.3 輸出層與預測

經過 {變換器 (Transformer) 後，序列中的每個位置都獲得了 128 維的特徵表示。為了得到整個字元的全局表示，本研究採用平均池化 (Average Pooling) 策略。

#### 全局特徵池化

對於長度為  $L$  的序列，編碼器輸出為  $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_L]$ ，其中  $\mathbf{h}_i \in \mathbb{R}^{128}$ 。全局特徵透過對所有有效位置的特徵取平均計算：

$$\mathbf{h}_{\text{global}} = \frac{1}{L_{\text{valid}}} \sum_{i=1}^{L_{\text{valid}}} \mathbf{h}_i \quad (4-41)$$

由於不同字元的線段數量不同，在批次處理時需要對較短的序列進行填充 (Padding)。為了避免填充位置影響平均池化結果，本研究在池化時應用填充遮罩 (Padding Mask)，僅對有效位置計算平均。

#### 分類頭

全局特徵  $\mathbf{h}_{\text{global}}$  輸入分類頭 (Classification Head)，預測字元類別的機率分布。分類頭為一個線性層，輸出  $C$  維的類別邏輯值 (logits)。對於本研究的工程圖字元辨識任務，類別包含數字 (0-9) 與特殊符號 (如「.」、「+」、「-」、「±」、「□」、「()」、「[]」等)。經過標籤過濾，實際使用的類別約為 20 類，包含 10 個數字與約 10 個特殊符號。

分類邏輯值透過 softmax 函數轉換為機率分布，訓練時使用交叉熵損失 (Cross-Entropy Loss)。

## 角度預測頭

工程圖中的字元可能以不同角度排列，準確預測字元的旋轉角度對於後續的字串組合至關重要。角度預測頭與分類頭並行，共享全局特徵  $\mathbf{h}_{\text{global}}$ 。

為了避免角度表示的周期性問題，角度預測頭輸出角度的正弦與餘弦值。透過線性層將 128 維全局特徵轉換為 2 維輸出  $[\hat{s}, \hat{c}]$ ，預測的角度值透過反正切函數還原：

$$\hat{\theta} = \text{atan2}(\hat{s}, \hat{c}) \quad (4-42)$$

其中  $\text{atan2}$  函數返回  $[-\pi, \pi]$  範圍內的角度，能正確處理四個象限的情況。

角度預測的損失函數將在 4.2.4 節詳細說明，因為需要考慮字元的對稱性特性。

### 4.2.4 多對稱性角度處理



工程圖中的字元具有不同的對稱性特徵，影響角度預測的目標範圍與損失計算。本研究根據字元的對稱性將其分為四類，並針對每類設計對應的角度處理策略。

## 對稱性分類

本研究根據字元的對稱性將其分為四類。第一類為完全  $360^\circ$  旋轉的字元，如數字「2」、「3」、「5」、「6」、「7」、「8」，以及大部分字母，這些字元不具有旋轉對稱性，角度範圍為  $[0^\circ, 360^\circ]$ 。第二類為  $180^\circ$  對稱字元，如數字「1」、「9」、字母「I」、「S」、「X」、「Z」，以及括號「()」、「[]」，這些字元旋轉  $180^\circ$  後外觀不變，角度  $\theta$  與  $\theta + 180^\circ$  等價，角度範圍可限制為  $[-90^\circ, 90^\circ]$ 。第三類為  $90^\circ$  對稱字元，如特殊符號「#」(井號)、「+」(加號)、「\*」(星號)，這些字元具有  $90^\circ$  旋轉對稱性，角度  $\theta$ 、 $\theta + 90^\circ$ 、 $\theta + 180^\circ$ 、 $\theta + 270^\circ$  均等價，角度範圍可限制為  $[-45^\circ, 45^\circ]$ 。第四類為完全對稱字元，如「.」(點)、「,」(逗號)、「o」(圓)，這些字元在所有角度下外觀完全相同，無需預測角度。表 4-1 總結四種對稱性類型與對應的字元範例。

圖 4.10: 相交組件合併範例。(a) 字元「E」的三個筆畫在預處理階段被分配到不同的連接組件 (用不同顏色標示)；(b) 進階點網路 (PointNet++) 逐點分類後，三個組件都被判定為文字；(c) 透過邊界框相交判定，三個組件被合併；(d) 最終輸出單一文字區域 (紅色矩形)。

圖 4.11: 字元辨識模組的完整架構。從連接線段輸入，經過特徵嵌入、變換器 (Transformer)、池化與輸出層，最終預測字元類別與旋轉角度。



圖 4.12: 字元「E」的筆畫排序。(a) 原始連接線段，紅點標示終端點 (度數為 1)；(b) 按照筆畫順序排列，箭頭表示線段方向，數字表示順序。可以看到筆畫從上到下、從左到右依次排列。

對稱性類型	角度範圍	字元範例	等價角度
完全 $360^\circ$	$0^\circ - 360^\circ$	2, 3, 5, 6, 7, 8, A-Z, a-z	無
$180^\circ$ 對稱	$-90^\circ - 90^\circ$	1, 9, I, S, X, Z, (), []	$\theta \equiv \theta + 180^\circ$
$90^\circ$ 對稱	$-45^\circ - 45^\circ$	#, +, *	$\theta \equiv \theta + 90^\circ$
完全對稱	不適用	., ,, °	所有角度等價

表 4-1: 字元的對稱性分類與角度範圍。不同對稱性的字元需要不同的角度預測策略。

## 對稱性適應的損失函數

對於具有對稱性的字元，直接使用均方誤差 (Mean Squared Error, MSE) 計算角度損失會導致訓練不穩定。例如，對於  $180^\circ$  對稱的字元「1」，真實角度為  $10^\circ$ ，如果模型預測  $190^\circ$  (等價於  $10^\circ$ )，標準 MSE 會認為誤差很大，但實際上預測是正確的。

為了解決此問題，本研究提出對稱性適應的角度損失函數。對於具有週期性對稱的字元，計算預測角度與所有等價角度的 MSE，取最小值作為最終損失。MSE 基於正弦與餘弦值計算，避免角度的週期性問題：

$$\text{MSE}(\hat{\theta}, \theta) = (\sin(\hat{\theta}) - \sin(\theta))^2 + (\cos(\hat{\theta}) - \cos(\theta))^2 \quad (4-43)$$

對於  $180^\circ$  對稱字元，預測角度  $\hat{\theta}$  與真實角度  $\theta^*$  的損失為：

$$\mathcal{L}_{\text{angle}}^{180} = \min \left( \text{MSE}(\hat{\theta}, \theta^*), \text{MSE}(\hat{\theta}, \theta^* + 180^\circ) \right) \quad (4-44)$$

對於  $90^\circ$  對稱字元，評估四個等價角度：

$$\mathcal{L}_{\text{angle}}^{90} = \min_{k \in \{0,1,2,3\}} \text{MSE}(\hat{\theta}, \theta^* + k \times 90^\circ) \quad (4-45)$$

對於無對稱性字元，使用標準 MSE：

$$\mathcal{L}_{\text{angle}}^{360} = \text{MSE}(\hat{\theta}, \theta^*) \quad (4-46)$$

在訓練時，根據每個字元的對稱性類型選擇對應的損失函數。總體角度損失為批次中所有字元的平均。

在評估階段，使用角度距離 (Angular Distance) 作為評估指標，計算預測角度與真實角度的最小夾角：

$$\text{Error}_{\text{angle}} = \min(|\hat{\theta} - \theta^*|, 360^\circ - |\hat{\theta} - \theta^*|) \quad (4-47)$$

對於具有對稱性的字元，同樣需要考慮等價角度，取所有等價角度中的最小誤差。

圖 4.13 展示不同對稱性字元的角度損失計算示意。

## 4.2.5 訓練策略

本節說明字元辨識模組的訓練配置與策略。

總體損失函數由分類損失與角度損失加權組合，角度損失的權重係數設定為 0.2：

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{cls}} + 0.2 \cdot \mathcal{L}_{\text{angle}} \quad (4-48)$$

這個權重設定平衡了分類與角度預測的重要性。在實驗中，過高的角度損失權重會導致模型過度關注角度預測而忽略分類準確率，過低則無法學習準確的角度資訊。權重 0.2 在驗證集上達到最佳的分類與角度預測平衡。

本研究採用 Adam 優化器，學習率設定為  $1 \times 10^{-4}$ 。較小的學習率避免了訓練初期的震盪，有利於變換器 (Transformer) 模型的穩定訓練。訓練時批次大小設定為 16。由於不同字元的線段數量差異較大，較小的批次大小減少了批次內序列長度的差異，降低了填充造成的計算浪費。

為了平衡訓練資料中不同類別的樣本數量，本研究採用類別均衡採樣 (Class-Balanced Sampling) 策略。每個訓練週期 (Epoch) 中，從每個類別隨機選擇 35 個樣本，確保所有類別都有足夠的訓練機會。這種策略避免了樣本數量多的類別主導訓練過程，提升了罕見類別的辨識準確率。

為了提升模型對旋轉的魯棒性，訓練時對每個樣本應用隨機旋轉增強。在  $[0^\circ, 360^\circ]$  範圍內隨機選擇一個角度  $\theta_{\text{aug}}$ ，將字元的所有線段座標旋轉  $\theta_{\text{aug}}$ ，並相應調整真實角度標籤。旋轉中心為字元的質心 (Centroid)，即所有線段端點的平均座標。這種增強策略使模型能學習旋轉不變的字元特徵，在推理階段面對不同角度的字元時表現更加穩定。

訓練過程中，在驗證集上監控分類準確率。當驗證集準確率連續 300 個 epoch 未提升時，停止訓練，避免過擬合。最大訓練週期數設定為 1000。在實驗中，大部分模型在 200-400 個 epoch 內收斂。

經過訓練後，字元辨識模組能夠準確預測字元的類別與旋轉角度。對於進階點網路 (PointNet++) 偵測到的文字區域，辨識模組輸出字元類別、角度資訊與邊界框座標，為後續的字串組合模組提供輸入。字串組合模組將在 4.3 節詳述。

## 4.3 兩階段配對模組

兩階段配對模組負責將字元辨識模組輸出的字串組合成有意義的標註單元。印刷電路板 (PCB) 中的標註資訊通常由多個字串組成，例如尺寸標註包含主數值、正公差與負公差，這些字串的排列方式與相對位置因格式而異。本模組採用兩階段策略完成組合任務：第一階段透過變換器 (Transformer) 網路預測候選字串對的配對可能性，快速篩選出可能屬於同一標註的字串對；第二階段基於規則對候選字串對進行精細配對，根據空間關係與格式特徵判斷字串的角色（主數值、正公差、負公差等），最終組合為完整的標註。圖 4.14 展示兩階段配對模組的完整流程。

相較於第三章的設計理念說明，本節將詳細闡述變換器 (Transformer) 配對網路的架構設計、特徵表示方式、訓練策略，以及基於規則的精細配對實作細節。

### 4.3.1 第一階段：基於變換器 (Transformer) 的配對網路



第一階段配對網路負責從所有可能的字串對中快速篩選出可能屬於同一標註的候選對。本網路基於變換器 (Transformer) 架構，接受兩個字串的特徵作為輸入，透過自注意力機制捕捉字串之間的關聯性，輸出配對的二元分類結果。本節將詳細說明字串對的特徵表示、變換器 (Transformer) 架構設計、訓練策略與損失函數。

#### 字串對的特徵表示

對於字元辨識模組輸出的字串集合  $\mathcal{S} = \{s_1, s_2, \dots, s_N\}$ ，其中每個字串  $s_i$  包含其類別序列、角度  $\theta_i$ 、邊界框座標  $(x_i^{\min}, y_i^{\min}, x_i^{\max}, y_i^{\max})$  與質心座標  $(c_x^i, c_y^i)$ 。對於字串對  $(s_i, s_j)$ ，本研究提取 21 維的配對特徵向量，包含空間關係特徵、類別特徵與格式特徵。

空間關係特徵描述兩個字串的相對位置與距離。首先，計算字串  $s_i$  與  $s_j$  的質心之間的距離：

$$d_{ij} = \sqrt{(c_x^j - c_x^i)^2 + (c_y^j - c_y^i)^2} \quad (4-49)$$

接著，計算相對方向向量並歸一化：

$$\Delta x = \frac{c_x^j - c_x^i}{d_{ij}}, \quad \Delta y = \frac{c_y^j - c_y^i}{d_{ij}} \quad (4-50)$$

相對角度  $\theta_{\text{rel}}$  定義為從字串  $s_i$  指向字串  $s_j$  的方向角：

$$\theta_{\text{rel}} = \text{atan2}(\Delta y, \Delta x) \quad (4-51)$$

為了避免角度的周期性問題，將相對角度編碼為正弦與餘弦值  $(\sin(\theta_{\text{rel}}), \cos(\theta_{\text{rel}}))$ 。

歸一化距離  $d_{\text{norm}}$  根據字串  $s_i$  的邊界框尺寸計算：

$$d_{\text{norm}} = \frac{d_{ij}}{\max(w_i, h_i)} \quad (4-52)$$

其中  $w_i = x_i^{\max} - x_i^{\min}$  為字串  $s_i$  的寬度， $h_i = y_i^{\max} - y_i^{\min}$  為高度。這種歸一化方式使距離特徵與字串尺寸相對應，避免絕對距離在不同尺度字串間的不一致。

類別特徵採用獨熱編碼 (One-Hot Encoding) 表示兩個字串的類別組合。本研究將字元類別分為四大類：數字 (Numeric)、點 (Dot)、加減號 (Plus/Minus)、其他符號 (Other)。對於字串  $s_i$  與  $s_j$ ，根據其主要類別（字串中出現最多的字元類別）確定所屬大類，並編碼為 16 維的獨熱向量。16 維對應四大類的兩兩組合，即  $(4 \times 4 = 16)$  種可能。例如，如果  $s_i$  為數字、 $s_j$  為點，則對應編碼的第 5 維（數字-點組合）為 1，其餘為 0。

格式特徵包含兩個字串的尺寸比例與角度差異。尺寸比例定義為字串  $s_j$  相對於  $s_i$  的面積比：

$$r_{\text{size}} = \frac{w_j \times h_j}{w_i \times h_i} \quad (4-53)$$

為了避免比例值過大，應用對數變換  $\log(1 + r_{\text{size}})$ 。角度差異定義為兩個字串角度的差值：

$$\Delta\theta = \theta_j - \theta_i \quad (4-54)$$

同樣編碼為正弦與餘弦值  $(\sin(\Delta\theta), \cos(\Delta\theta))$ 。

將以上特徵串接，形成 21 維的配對特徵向量：

$$\mathbf{f}_{ij} = [d_{\text{norm}}, \sin(\theta_{\text{rel}}), \cos(\theta_{\text{rel}}), \text{onehot}_{16}, \log(1 + r_{\text{size}}), \sin(\Delta\theta), \cos(\Delta\theta)] \quad (4-55)$$

其中  $\text{onehot}_{16}$  為 16 維的類別組合獨熱編碼。這種特徵表示涵蓋了空間關係、類別資訊與格式差異，為變換器 (Transformer) 提供了豐富的配對判斷依據。

### 變換器 (Transformer) 配對網路架構

對於每對候選字串  $(s_i, s_j)$ ，配對網路的目標是預測它們是否屬於同一標註。本研究採用變換器 (Transformer) Encoder 架構，透過自注意力機制捕捉特徵之間的關聯性，最終輸出二元分類結果。

首先，將 21 維的配對特徵  $\mathbf{f}_{ij}$  透過線性層投影為 128 維的嵌入向量  $\mathbf{h}_0$ 。投影層採用全連接網路，包含一層線性變換、批次歸一化與 ReLU 激活函數：



$$\mathbf{h}_0 = \text{ReLU}(\text{BN}(\mathbf{W}_{\text{proj}}\mathbf{f}_{ij} + \mathbf{b}_{\text{proj}})) \quad (4-56)$$

其中  $\mathbf{W}_{\text{proj}} \in \mathbb{R}^{128 \times 21}$  為投影權重矩陣， $\mathbf{b}_{\text{proj}} \in \mathbb{R}^{128}$  為偏置向量。

{變換器 (Transformer) 由 2 層堆疊而成，每層包含多頭自注意力機制與前饋網路。隱藏維度設定為 128，採用 4 個注意力頭，每個頭的維度為 32。前饋網路的隱藏層維度為 256，即透過兩層全連接網路，將維度從 128 擴展到 256 再還原為 128，中間使用 ReLU 激活函數。每個子層後使用殘差連接與層歸一化穩定訓練過程，並應用 Dropout (丟棄率 0.1) 防止過擬合。

由於配對特徵為單一向量而非序列，變換器 (Transformer) 的輸入序列長度為 1。在這種情況下，自注意力機制退化為對特徵本身的加權變換，但前饋網路仍能提取高層次的非線性特徵。透過 2 層的堆疊，模型能學習特徵之間的複雜組合模式。

經過 {變換器 (Transformer) 後，輸出 128 維的特徵  $\mathbf{h}_2$ 。最終透過分類頭 (Classification Head) 預測配對機率。分類頭為一個線性層，將 128 維特徵轉換為 2 維的邏輯值 (logits)，分別對應「不配對」與「配對」兩個類別：

$$\text{logits} = \mathbf{W}_{\text{cls}}\mathbf{h}_2 + \mathbf{b}_{\text{cls}} \quad (4-57)$$

其中  $\mathbf{W}_{cls} \in \mathbb{R}^{2 \times 128}$  為分類權重矩陣。透過 softmax 函數將邏輯值轉換為機率分布：

$$p_{pair} = \text{softmax}(\text{logits})[1] \quad (4-58)$$

其中  $p_{pair}$  為配對機率。當  $p_{pair} > \tau_{conf}$  時，判定字串對  $(s_i, s_j)$  為候選配對，其中信心閾值  $\tau_{conf} = 0.5$ 。

圖 4.15 展示變換器 (Transformer) 配對網路的完整架構。

### 訓練資料準備

配對網路的訓練需要大量正樣本（屬於同一標註的字串對）與負樣本（不屬於同一標註的字串對）。本研究基於標註資料生成訓練樣本。

在標註階段，標註人員對每個完整標註標記包含哪些字串。例如，尺寸標註「1.5 +0.1 -0.0」被標記為包含三個字串：主數值「1.5」、正公差「+0.1」與負公差「-0.0」。基於這些標註，同一標註內的任意兩個字串構成正樣本對。

對於負樣本，本研究採用隨機採樣策略與難例挖掘 (Hard Negative Mining) 相結合的方法。隨機採樣階段，對於每個標註，從其他標註中隨機選擇字串，與該標註的字串組成負樣本對。這種策略能生成大量負樣本，涵蓋各種空間關係與類別組合。

難例挖掘階段，針對容易被誤判為配對的字串對進行額外採樣。具體而言，對於距離較近 ( $d_{ij} < 2 \times \max(w_i, h_i)$ ) 但不屬於同一標註的字串對，增加其在訓練資料中的比例。這種策略使模型能學習區分空間上相近但實際不配對的字串，提升判斷的精確度。

為了平衡正負樣本的數量，每個訓練批次中正負樣本的比例設定為 1:2。由於工程圖中大部分字串對不配對，負樣本的數量遠多於正樣本。透過控制採樣比例，避免模型傾向預測所有字串對為不配對。

### 損失函數與訓練策略

訓練時使用加權交叉熵損失 (Weighted Cross-Entropy Loss)。由於正負樣本數量的不平衡，對正樣本賦予較高的權重  $w_{pos} = 2.0$ ，負樣本權重  $w_{neg} = 1.0$ ：

$$\mathcal{L} = -\frac{1}{B} \sum_{k=1}^B [y_k w_{\text{pos}} \log(p_k) + (1 - y_k) w_{\text{neg}} \log(1 - p_k)] \quad (4-59)$$

其中  $B$  為批次大小， $y_k \in \{0, 1\}$  為真實標籤（1 表示配對，0 表示不配對）， $p_k$  為模型預測的配對機率。這種加權策略確保模型不會忽略數量較少的正樣本，提升召回率。

訓練配置方面，批次大小設定為 8，採用 Adam 優化器，初始學習率為  $1 \times 10^{-3}$ 。學習率調度採用分段策略：前 40 個 epoch 為預熱階段 (Warm-up)，學習率從 0 線性增長至初始值  $1 \times 10^{-3}$ ；第 41 到 80 個 epoch 保持恆定；從第 81 個 epoch 開始，每 20 個 epoch 將學習率乘以 0.5，下限設定為  $1 \times 10^{-5}$ 。這種調度策略使模型在訓練初期能夠平穩學習，後期透過降低學習率精細調整參數。

資料增強方面，對於每個訓練樣本，在計算配對特徵前，對兩個字串的座標施加小幅度的隨機擾動。擾動幅度為字串尺寸的 5%，即  $\Delta c_x \sim \mathcal{U}(-0.05w, 0.05w)$ ， $\Delta c_y \sim \mathcal{U}(-0.05h, 0.05h)$ 。這種增強策略模擬了字元辨識模組可能產生的位置誤差，提升配對網路對位置雜訊的魯棒性。

訓練過程中，在驗證集上監控 F1 分數。當驗證集 F1 分數連續 20 個 epoch 未提升時，停止訓練，避免過擬合。最大訓練週期數設定為 200。在實驗中，大部分模型在 100-150 個 epoch 內收斂。

經過訓練後，配對網路能夠準確預測字串對的配對可能性。對於字元辨識模組輸出的  $N$  個字串，配對網路評估所有  $\binom{N}{2} = \frac{N(N-1)}{2}$  個可能的字串對，篩選出配對機率超過閾值的候選對。這些候選對將輸入第二階段的精細配對模組，進行角色判斷與最終組合。第二階段的詳細實作將在 4.3.2 節闡述。

圖??展示第一階段配對網路在測試集上的配對結果範例。可以看到，網路能成功識別屬於同一標註的字串對，同時過濾掉大量不相關的字串對，大幅減少第二階段需要處理的候選數量。

圖 4.13: 不同對稱性字元的角度損失計算。(a) 字元「2」(無對稱性)，僅評估單一角度；(b) 字元「1」( $180^\circ$  對稱)，評估  $\theta$  與  $\theta + 180^\circ$  兩個等價角度；(c) 字元「+」( $90^\circ$  對稱)，評估四個等價角度。綠色箭頭表示真實角度，紅色箭頭表示預測角度，藍色箭頭表示等價角度。



圖 4.14: 兩階段配對模組的完整流程。第一階段：變換器 (Transformer) 網路預測候選字串對的配對機率；第二階段：基於規則的精細配對判斷字串角色與組合方式。

圖 4.15: 變換器 (Transformer) 配對網路的完整架構。從 21 維配對特徵輸入，經過投影層、2 層 {變換器 (Transformer)、分類頭，最終輸出配對機率。

# 第 5 章 實驗設計與結果討論

為了驗證本研究提出的基於線段連接關係的點雲偵測方法與兩階段自動配對方法的有效性，本章節設計了三個層次的實驗。首先，透過消融測試驗證各項設計決策的有效性，包括改進的採樣策略、改進的分組策略與斜率特徵編碼。接著，將本研究的文字辨識方法與商業光學字元辨識 (OCR) 系統及學術方法比較，驗證在遮擋場景下的優勢。最後，將字串組合方法與基於規則的方法比較，驗證自動學習配對策略的可行性與格式適應能力。透過定量指標、分類評估與視覺化案例分析，全面評估本研究方法的準確率與實用性。

## 5.1 實驗基礎

本節介紹實驗所使用的資料集與評估指標。資料集的設計考慮了實際應用場景的多樣性，包含不同來源的工程圖、不同程度的遮擋情況與不同的標註格式。評估指標的選擇參考現有工程圖自動化系統的評估方式，同時針對本研究的核心貢獻設計專項指標。



### 5.1.1 資料集

本研究使用的資料集來自英業達公司提供的印刷電路板佈局圖 (P.C.B. Layout) 工程圖，涵蓋多種產品類型與設計風格，確保資料的多樣性與代表性。資料集分為訓練集與測試集，測試集進一步分為整體測試集、遮擋場景測試集與多格式測試集，以全面評估方法的性能。

#### 訓練資料集

訓練資料集包含 357 張印刷電路板佈局圖 (P.C.B. Layout) 工程圖，經過人工標註後產生 36897 個字元標註與 5498 個標註組。標註過程中，標註人員標記每個字元的位置、類別與旋轉角度，並標記字串之間的組合關係。為了確保訓練資料的品質，所有標註經過 2 輪交叉驗證與修正。

訓練資料的特性包括：第一，包含 300 個遮擋案例，並在訓練中，其中字元被針腳 (Pin) 遮擋佔 XXX%，被線段遮擋佔 XXX%，確保模型能學習處理遮擋場景。第

二，包含 XXX 種不同的標註格式，涵蓋上下標結構、對稱表示（如土）與平行排列等常見格式。第三，字元類別分布涵蓋數字、英文字母、符號等 XXX 種類別，其中數字佔 XXX%，字母佔 XXX%，符號佔 XXX%。

## 測試資料集

測試資料集的來源與訓練資料集不同，確保評估結果能反映方法的泛化能力。測試資料集包含 88 張工程圖，產生 21011 個字元標註與 3131 個標註組。

為了評估方法在不同場景下的性能，測試資料集進一步分為以下三類：

第一，整體測試集包含所有 3012 個字串，用於評估方法的整體準確率。

第二，遮擋場景測試集將字串分為被遮擋字串與未遮擋字串。被遮擋字串定義為字串中至少有一個字元與其他元件或線段在空間上重疊，包含 75 個字串，佔整體測試集的 7%。未遮擋字串定義為字串中所有字元均無遮擋，包含 2937 個字串。透過分類評估，能夠量化方法在遮擋場景下的優勢。



第三，多格式測試集將標註組按格式分類，包含 XXX 個上下標結構標註、XXX 個對稱表示標註與 XXX 個平行排列標註。部分標註格式在訓練資料中出現頻率較低，用於測試方法對新格式的適應能力。

### 5.1.2 評估指標

本研究採用字串準確率作為文字辨識的主要評估指標，採用配對準確率與標註提取準確率作為字串組合的評估指標。指標的選擇參考托羅等人 (Toro et al.) 等人 [1] 的評估方式，同時針對本研究的核心貢獻設計分類評估。

#### 文字辨識指標

字串準確率定義為正確辨識的字串數量與總字串數量的比值：

$$\text{字串準確率} = \frac{\text{正確辨識的字串數量}}{\text{總字串數量}} \quad (5-1)$$

其中，正確辨識的定義為字串中所有字元的類別與位置均正確。此指標直接反映方法的實用價值，因為在實際應用中，標註資訊需要完整正確才能用於自動化製程。

為了評估方法在不同場景下的表現，本研究進一步計算被遮擋字串準確率與未遮擋字串準確率。被遮擋字串準確率定義為正確辨識的被遮擋字串數量與總被遮擋字串數量的比值，未遮擋字串準確率定義類似。透過比較兩者的差異，能夠量化方法在遮擋場景下的優勢。

此外，本研究也記錄字元準確率作為輔助指標，定義為正確辨識的字元數量與總字元數量的比值。字元準確率能提供更細緻的性能分析，例如識別哪些字元類別容易混淆。

### 字串組合指標

配對準確率定義為正確配對的字串組數量與總字串組數量的比值：

$$\text{配對準確率} = \frac{\text{正確配對的字串組數量}}{\text{總字串組數量}} \quad (5-2)$$

其中，正確配對的定義為字串組內所有字串的歸屬關係均正確。例如，尺寸標註包含主數值、正公差與負公差三個字串，當且僅當三個字串均被正確歸類到同一個字串組時，該字串組被判定為正確配對。

完整標註提取準確率定義為正確提取的完整標註數量與總標註數量的比值：

$$\text{標註提取準確率} = \frac{\text{正確提取的完整標註數量}}{\text{總標註數量}} \quad (5-3)$$

此指標結合文字辨識與字串組合兩個階段，反映端到端系統的性能。當且僅當標註中所有字元均被正確辨識，且所有字串均被正確配對時，該標註被判定為正確提取。

### 指標選擇理由

本研究選擇字串準確率而非字元準確率作為主要指標，原因在於製造業的實際應用需求。在自動化製程中，部分錯誤的標註可能導致設備定位錯誤或品質檢驗失敗，因此標註資訊需要完整正確。字串準確率能直接反映這種需求，相較於字元準確率更具實用意義。

此外，本研究採用分類評估的方式，分別計算被遮擋場景與未遮擋場景的準確率。這種評估方式能夠凸顯本研究方法在遮擋場景下的核心優勢，同時確保方法在一般場景下的性能不遜於現有方法。

配對準確率的設計則參考文件理解領域的版面分析評估方式，關注字串之間的關聯關係而非字串本身內容。透過配對準確率，能夠獨立評估兩階段配對方法的有效性，排除文字辨識誤差的影響。

## 5.2 消融測試

本節透過消融測試驗證各項設計決策的有效性。消融測試的設計原則為控制變因，每次僅改變一個設計元素，觀察其對整體性能的影響。本節包含三個消融測試：改進的採樣策略、改進的分組策略與斜率特徵編碼。所有消融測試均在相同的訓練資料與測試資料上進行，確保結果的可比性。



### 5.2.1 改進的採樣策略有效性

本測試驗證第 4.2 節提出的線段組感知最遠點取樣 (Line Group-Aware FPS) 的有效性。標準的最遠點取樣 (FPS) 僅考慮點的空間分布，未考慮連接組件的代表性。當字元包含多個線段組成的筆畫時，標準最遠點取樣 (FPS) 可能在某些筆畫上過度採樣，而在其他筆畫上採樣不足，導致模型在特徵聚合時缺乏足夠的組件資訊。線段組感知最遠點取樣 (Line Group-Aware FPS) 透過追蹤每個連接組件的採樣數量，確保每個組件都有最少的代表點。

實驗設置包含兩個方法：Baseline 採用標準最遠點取樣 (FPS)，在進階點網路 (PointNet++) 的第一層集合抽象層 (Set Abstraction, SA) 中選擇 512 個採樣點；Ours 採用線段組感知最遠點取樣 (Line Group-Aware FPS)，同樣選擇 512 個採樣點，但在採樣過程中追蹤連接組件的採樣數量，確保每個組件至少採樣 2 個點。兩個方法的其他設置 (分組策略、網路架構、訓練參數) 均保持一致。

表 5-1 展示改進的採樣策略對字串辨識準確率的影響。結果顯示，線段組感知最遠點取樣 (Line Group-Aware FPS) 在整體字串準確率上提升 XXX%，在被遮擋字串準確率上提升 XXX%，在未遮擋字串準確率上提升 XXX%。這證明追蹤連接組件的採樣數量能夠提升模型對字元結構的理解，特別是對包含多個筆畫的複雜字元。

進一步分析顯示，線段組感知最遠點取樣 (Line Group-Aware FPS) 對包含較多線段的字元改善更為顯著。例如，字元「W」、「M」等通常由 XXX 條以上的線段組成，標準最遠點取樣 (FPS) 可能僅在部分線段上採樣，導致模型缺乏完整的字元結構資訊。線段組感知最遠點取樣 (Line Group-Aware FPS) 確保每條線段都有代表點，使模型能獲取完整的字元形狀特徵。

### 5.2.2 改進的分組策略有效性

本測試驗證第 4.3 節提出的雙重分組策略的有效性。標準的進階點網路 (PointNet++) 採用多尺度分組 (MSG)，以多個半徑進行球形查詢 (Ball Query)，捕捉不同尺度的幾何特徵。然而，在遮擋場景下，球形查詢 (Ball Query) 僅依據歐氏距離選點，無法區分相連線段與遮擋物件的點，導致特徵混雜。本研究提出的雙重分組策略在多尺度分組 (MSG) 的基礎上，引入基於連接關係的同組查詢 (Same-Group Query)，優先聚合相連線段的點。

實驗設置包含三個方法：Baseline 1 僅使用單一尺度的球形查詢 (Ball Query) (半徑 0.1，選取 32 個鄰居)，作為最基礎的比較基準；Baseline 2 採用標準多尺度分組 (MSG) (三個尺度：半徑 0.1 選取 32 個鄰居、半徑 0.05 選取 64 個鄰居、半徑 0.1 選取 128 個鄰居)，所有尺度均使用球形查詢 (Ball Query)；Ours 採用雙重分組策略，前兩個尺度使用球形查詢 (Ball Query)，第三個尺度使用同組查詢 (Same-Group Query)，僅選取與中心點屬於相同連接組件的點。三個方法的其他設置均保持一致。

表 5-2 展示改進的分組策略對字串辨識準確率的影響。結果顯示，標準多尺度分組 (MSG) 相較於單一尺度球形查詢 (Ball Query) 在整體準確率上提升 XXX%，證明多尺度特徵的有效性。然而，標準多尺度分組 (MSG) 在被遮擋場景下的改善有限 (僅提升 XXX%)，因為遮擋處的點仍會混雜不同物件的資訊。本研究的雙重分組策略在被遮擋字串準確率上顯著提升 XXX%，證明同組查詢 (Same-Group Query) 能有效排除遮擋物件的干擾，使模型在遮擋處獲取正確的字元結構特徵。

圖 5.1 展示在遮擋場景下，三種分組策略的特徵聚合差異。以字元「E」被線段遮擋為例，單一尺度球形查詢 (Ball Query) 僅能捕捉局部特徵，在遮擋處選取的鄰居包含遮擋線段的點，導致特徵污染。標準多尺度分組 (MSG) 透過多尺度特徵能夠捕捉更廣範圍的資訊，但在遮擋處仍存在相同問題。本研究的雙重分組策略在第三尺度使用同組查詢 (Same-Group Query)，僅選取與字元「E」屬於相同連接組件的點，成功排除遮擋線段的干擾，使模型能在遮擋處獲取足夠的字元結構特徵。

方法	整體字串準確率	被遮擋字串準確率	未遮擋字串準確率
標準最遠點取樣 (FPS)	XXX%	XXX%	XXX%
線段組感知最遠點取樣 (Line Group-Aware FPS) (Ours)	XXX%	XXX%	XXX%
改善	+XXX%	+XXX%	+XXX%

表 5-1: 改進的採樣策略對字串辨識準確率的影響。線段組感知最遠點取樣 (Line Group-Aware FPS) 透過追蹤連接組件的採樣數量，確保每個組件都有足夠的代表點，在各項指標上均優於標準最遠點取樣 (FPS)。

方法	整體字串準確率	被遮擋字串準確率	未遮擋字串準確率
單一度球形查詢 (Ball Query)	XXX%	XXX%	XXX%
標準多尺度分組 (MSG)	XXX%	XXX%	XXX%
雙重分組 (Ours)	XXX%	XXX%	XXX%

表 5-2: 改進的分組策略對字串辨識準確率的影響。雙重分組策略結合標準球形查詢 (Ball Query) 與同組查詢 (Same-Group Query)，在遮擋場景下顯著優於標準多尺度分組 (MSG)，在未遮擋場景下維持相當性能。

圖 5.1: 遮擋場景下不同分組策略的特徵聚合差異。(a) 字元「E」被線段遮擋，紅色圓圈標示遮擋位置；(b) 單一度球形查詢 (Ball Query) 在遮擋處聚合了遮擋線段的點 (紅色標記)；(c) 標準多尺度分組 (MSG) 仍存在特徵混雜問題；(d) 雙重分組策略的同組查詢 (Same-Group Query) 成功排除遮擋物，僅聚合字元本身的點 (綠色標記)。

值得注意的是，雙重分組策略在未遮擋場景下的準確率與標準多尺度分組 (MSG) 相當，證明同組查詢 (Same-Group Query) 不會犧牲一般場景的性能。這是因為在未遮擋場景下，同組查詢 (Same-Group Query) 選取的點與球形查詢 (Ball Query) 選取的點高度重疊，連接組件的限制不會排除有用的鄰居。因此，雙重分組策略能在遮擋場景下提供顯著優勢，同時在一般場景下維持穩定性能。

### 5.2.3 斜率特徵編碼的影響

本測試驗證第 4.1.3 節提出的雙角度轉換斜率特徵編碼的有效性。線段的斜率是判斷字元結構的重要特徵，特別是在辨識具有明確筆畫方向的字元時。然而，直接使用角度  $\theta \in [0, 2\pi)$  表示斜率會遇到周期性問題，例如  $\theta = 0^\circ$  與  $\theta = 180^\circ$  在視覺上表示相同方向但在數值上相差  $180^\circ$ 。本研究採用雙角度轉換去除這種方向模糊性，將斜率編碼為  $(\cos(2\theta'), \sin(2\theta'), 0)$ 。

實驗設置包含三個方法：Baseline 1 不使用斜率特徵，每個點的特徵僅包含座標  $(x, y, z)$  與線段長度  $L$ ；Baseline 2 使用原始角度  $\theta$ ，將其編碼為  $(\cos \theta, \sin \theta, 0)$ ；Ours 使用雙角度轉換，將斜率編碼為  $(\cos(2\theta'), \sin(2\theta'), 0)$ ，其中  $\theta' \in [-\pi/2, \pi/2]$ 。三個方法的其他設置均保持一致。

表 5-3 展示斜率特徵編碼對字串辨識準確率的影響。結果顯示，不使用斜率特徵的方法準確率最低，證明斜率資訊對字元辨識的重要性。使用原始角度的方法相較於無斜率特徵提升 XXX%，但仍受到方向模糊性的影響。本研究的雙角度轉換方法在整體準確率上進一步提升 XXX%，證明去除方向模糊性能幫助模型學習更穩定的特徵表示。

進一步分析顯示，斜率特徵對具有明確筆畫方向的字元影響更為顯著。例如，字元「E」、「F」、「H」等包含水平與垂直筆畫，斜率特徵能幫助模型區分不同方向的線段。對於這類字元，雙角度轉換方法的準確率提升達 XXX%。相較之下，對於圓形字元（如「O」、「Q」）或斜向字元（如「X」），斜率特徵的影響較小，因為這些字元的線段方向較為多樣，斜率特徵的區分能力有限。

## 5.3 文字辨識比較實驗

本節將本研究的文字辨識方法與商業光學字元辨識 (OCR) 系統及學術方法比較，驗證方法在整體準確率與遮擋場景下的優勢。比較方法包括谷歌光學字元辨識 (Google OCR)

與 (CRAFT) [4]，兩者均為廣泛應用的文字辨識系統。為了公平比較，所有方法均在相同的測試資料集上評估，採用相同的評估指標。透過整體辨識準確率比較、遮擋場景專項比較與視覺化案例分析，全面評估本研究方法的性能。

### 5.3.1 整體辨識準確率比較

本測試比較本研究方法與現有方法在整體測試集上的字串辨識準確率。谷歌光學字元辨識 (Google OCR) 是谷歌提供的商業光學字元辨識 (OCR) 服務，採用深度學習技術，在一般文件與場景文字辨識中表現優異。(CRAFT) [4] 是一種基於深度學習的文字偵測方法，透過預測字元級別的熱圖 (Heatmap) 與連接熱圖 (Affinity map) 定位文字區域，結合辨識模型完成端到端的文字辨識。

由於谷歌光學字元辨識 (Google OCR) 與 (CRAFT) 均為點陣圖光學字元辨識 (OCR) 方法，需要將向量格式的工程圖轉換為點陣圖。本研究將工程圖轉換為 300 DPI 的點陣圖，此解析度能在保持圖像細節的同時避免過大的檔案大小。所有方法均在相同的測試資料集上評估，採用字串準確率作為評估指標。

表 5-4 展示三種方法在整體測試集、被遮擋字串測試集與未遮擋字串測試集上的準確率。結果顯示，本研究方法在整體字串準確率上達到 XXX%，優於谷歌光學字元辨識 (Google OCR) 的 XXX% 與 (CRAFT) 的 XXX%。在被遮擋字串準確率上，本研究方法達到 XXX%，顯著優於谷歌光學字元辨識 (Google OCR) 的 XXX% 與 (CRAFT) 的 XXX%，改善幅度達 XXX%。在未遮擋字串準確率上，本研究方法達到 XXX%，與谷歌光學字元辨識 (Google OCR) 的 XXX% 相當，略優於 (CRAFT) 的 XXX%。

這些結果證明本研究方法在遮擋場景下具有顯著優勢，同時在一般場景下的性能不遜於商業系統。在被遮擋場景下的優勢源於本研究方法保留了向量資料中的線段連接關係與斜率資訊，透過同組查詢 (Same-Group Query) 優先聚合相連線段的點，使模型能在遮擋處獲取正確的字元結構特徵。相較之下，點陣圖方法在點陣化過程中將重疊線段合併為單一像素區域，丟失了線段的獨立性與連接資訊，導致遮擋處難以辨識。

在未遮擋場景下，本研究方法與商業光學字元辨識 (OCR) 系統的性能相當，證明向量資料的處理方式不會犧牲一般場景的準確率。谷歌光學字元辨識 (Google OCR) 作為商業系統，在大規模訓練資料與模型優化方面具有優勢，但在遮擋場景下受限於點陣化的固有問題。本研究方法透過直接處理向量資料，避免了這種限制，在保持一般場景性能的同時提升遮擋場景的準確率。

### 5.3.2 遮擋場景專項比較

為了進一步量化本研究方法在遮擋場景下的優勢，本測試專門分析被遮擋字串的辨識結果。測試集包含 XXX 個被遮擋字串，涵蓋不同程度與類型的遮擋情況，包括字元被針腳 (Pin) 遮擋、字元被線段遮擋、字元被其他元件遮擋等。

表 5-5 展示三種方法在被遮擋字串測試集上的詳細結果。本研究方法正確辨識 XXX 個被遮擋字串，準確率達 XXX%。谷歌光學字元辨識 (Google OCR) 正確辨識 XXX 個，準確率為 XXX%。(CRAFT) 正確辨識 XXX 個，準確率為 XXX%。本研究方法相較於谷歌光學字元辨識 (Google OCR) 改善 XXX 個字串，相較於 (CRAFT) 改善 XXX 個字串，證明在遮擋場景下的顯著優勢。

點陣圖光學字元辨識 (OCR) 方法在遮擋場景下失敗的主要原因是點陣化過程導致的資訊損失。當字元與其他元件或線段重疊時，點陣化將所有線段投影至同一平面並轉換為像素，重疊區域的像素值由所有線段共同決定。這導致兩個問題：第一，遮擋處的字元線段與遮擋物線段在視覺上合併，光學字元辨識 (OCR) 系統無法區分哪些像素屬於字元、哪些像素屬於遮擋物；第二，遮擋可能改變字元的視覺外觀，例如橫線與豎線在遮擋處連接，導致光學字元辨識 (OCR) 系統將字元誤判為其他類別。

本研究方法透過直接處理向量資料，保留了線段的獨立性。在預處理階段，每條線段的端點連接關係被記錄並用於連接分組。在分組階段，同組查詢 (Same-Group Query) 限制只選取與中心點屬於相同連接組件的點，即使在遮擋處多條線段的點在空間上混雜，模型仍能優先聚合屬於同一字元的相連線段點。此外，斜率特徵提供了額外的區分能力，幫助模型識別不同方向的線段。透過這些設計，本研究方法能在遮擋處獲取足夠的字元結構特徵，成功辨識被遮擋的字元。

### 5.3.3 視覺化案例分析

為了更直觀地展示本研究方法在遮擋場景下的優勢，本節提供視覺化案例分析。圖 5.2 展示兩個典型的成功案例，比較本研究方法與點陣圖光學字元辨識 (OCR) 方法的辨識結果。

第一個案例展示元件編號被針腳 (Pin) 遮擋的情況。如圖 5.2(a) 所示，字串「R10」中的字元「1」被針腳 (Pin) 部分遮擋。谷歌光學字元辨識 (Google OCR) 在點陣化後，遮擋處的線段合併，將字元「1」誤判為「I」，輸出結果為「RI0」。

方法	整體字串準確率	被遮擋字串準確率	未遮擋字串準確率
無斜率特徵	XXX%	XXX%	XXX%
原始角度 $\theta$	XXX%	XXX%	XXX%
雙角度轉換 (Ours)	XXX%	XXX%	XXX%

表 5-3: 斜率特徵編碼對字串辨識準確率的影響。雙角度轉換透過去除方向模糊性，使模型能學習更穩定的斜率特徵，在各項指標上均優於其他方法。

方法	整體字串準確率	被遮擋字串準確率	未遮擋字串準確率
谷歌光學字元辨識 ( Google OCR )	XXX%	XXX%	XXX%
(CRAFT) [4]	XXX%	XXX%	XXX%
本研究方法	XXX%	XXX%	XXX%

表 5-4: 本研究方法與商業光學字元辨識 (OCR) 系統及學術方法的字串辨識準確率比較。本研究方法在整體準確率與被遮擋場景下均優於現有方法，在未遮擋場景下維持相當性能。

方法	被遮擋字串總數	正確辨識數量	準確率
谷歌光學字元辨識 ( Google OCR )	XXX	XXX	XXX%
(CRAFT) [4]	XXX	XXX	XXX%
本研究方法	XXX	XXX	XXX%
改善幅度 ( vs 谷歌光學字元辨識 ( Google OCR ) )	-	+XXX	+XXX%
改善幅度 ( vs (CRAFT) )	-	+XXX	+XXX%

表 5-5: 本研究方法與現有方法在被遮擋字串測試集上的詳細比較。本研究方法在遮擋場景下顯著優於點陣圖光學字元辨識 (OCR) 方法，證明保留向量資料的線段連接關係與幾何資訊的有效性。

(CRAFT) 同樣受到點陣化的影響，未能正確偵測遮擋處的字元，輸出結果為「R?0」。本研究方法透過同組查詢 (Same-Group Query) 優先聚合字元「1」相連的線段點，成功排除針腳 (Pin) 的干擾，正確輸出「R10」。

第二個案例展示尺寸標註被箭頭線段遮擋的情況。如圖 5.2(b) 所示，字串「1.5 +0.1 -0.0」中的字元「+」與「-」被箭頭線段遮擋。谷歌光學字元辨識 (Google OCR) 在遮擋處無法辨識符號，輸出結果為「1.5 0.1 0.0」，遺漏了公差的正負符號。(CRAFT) 偵測到遮擋處的符號但辨識錯誤，輸出結果為「1.5 =0.1 -0.0」。本研究方法正確辨識所有字元與符號，輸出「1.5 +0.1 -0.0」。

這些案例證明本研究方法在遮擋場景下能有效處理線段重疊的問題，透過保留向量資料的幾何資訊與連接關係，克服點陣圖方法的固有限制。在實際應用中，工程圖上的遮擋情況普遍存在，本研究方法能顯著提升自動化標註系統的可靠性。

## 5.4 字串組合比較實驗



本節將本研究的兩階段自動配對方法與基於規則的方法比較，驗證自動學習配對策略的可行性與格式適應能力。比較方法包括托羅等人 (Toro et al.) 等人 [1,2] 提出的工程圖自動化標註系統，該系統採用基於規則的配對方法，透過空間關係與格式特徵設計規則。實驗包含端到端系統比較、配對策略專項比較與不同標註格式適應性分析，透過定量指標與案例分析全面評估本研究方法的性能。

### 5.4.1 端到端系統比較

本測試比較本研究的端到端系統與 (CRAFT) 結合基於規則配對方法的系統。(CRAFT) [4] 用於文字辨識，托羅等人 (Toro et al.) 等人 [1] 提出的規則方法用於字串組合。本研究系統則包含字元偵測、字元識別與兩階段配對三個模組，實現從向量工程圖輸入到標註資訊輸出的完整流程。

評估指標採用完整標註提取準確率，定義為正確提取的完整標註數量與總標註數量的比值。當且僅當標註中所有字元均被正確辨識且所有字串均被正確配對時，該標註被判定為正確提取。此指標反映端到端系統在實際應用中的性能，因為製造業需要完整正確的標註資訊才能用於自動化製程。

表 5-6 展示兩種系統在整體測試集上的完整標註提取準確率。結果顯示，本研究系統達到 XXX% 的準確率，優於 (CRAFT) 結合規則方法的 XXX%。改善幅度達 XXX%，證明本研究在文字辨識與字串組合兩方面的綜合優勢。

端到端準確率的改善源於兩個層次的貢獻。第一，在文字辨識階段，本研究方法在遮擋場景下的優勢減少了辨識錯誤傳播到後續配對階段的可能性。當字元辨識錯誤時，即使配對策略正確，最終的標註提取仍會失敗。本研究方法透過提升辨識準確率，為後續配對提供更可靠的輸入。第二，在字串組合階段，本研究的兩階段自動配對方法能適應不同的標註格式，相較於基於規則的方法具有更好的泛化能力。

#### 5.4.2 配對策略專項比較

為了排除文字辨識誤差的影響，純粹比較配對策略的性能，本測試採用 Ground Truth 辨識結果作為輸入。具體而言，兩種方法均使用人工標註的字元位置、類別與角度，僅在字串組合階段採用不同的策略。基於規則的方法採用托羅等人 (Toro et al.) 等人 [1] 提出的規則，透過字串之間的距離、相對位置與字體大小比例判斷配對關係。本研究方法採用兩階段自動配對，透過學習語意特徵與空間特徵進行配對。

評估指標採用配對準確率，定義為正確配對的字串組數量與總字串組數量的比值。此指標獨立評估配對策略的有效性，不受辨識誤差影響。

表 5-7 展示兩種配對策略在 Ground Truth 辨識結果上的配對準確率。結果顯示，本研究的兩階段自動配對方法達到 XXX% 的準確率，與基於規則方法的 XXX% 相當，差異僅為 XXX%。這證明自動學習的配對策略能達到與人工設計規則同等的性能，驗證了第三章提出的設計理念的可行性。

值得注意的是，雖然兩種方法在整體準確率上相當，但在不同標註格式上的表現存在差異。基於規則的方法針對常見格式（如上下標結構）設計規則，在這些格式上表現優異。然而，當遇到訓練時未充分考慮的格式時，規則可能失效。本研究的自動配對方法透過學習訓練資料中的模式，能夠適應不同格式，在格式多樣性方面具有優勢。下一節將詳細分析這種適應性。

圖 5.2: 本研究方法在遮擋場景下的成功案例。(a) 元件編號「R10」被針腳 (Pin) 遮擋，點陣圖光學字元辨識 (OCR) 方法因線段合併而失敗，本研究方法透過保留線段連接關係成功辨識；(b) 尺寸標註「1.5 +0.1 -0.0」被箭頭線段遮擋，點陣圖光學字元辨識 (OCR) 方法遺漏或誤判符號，本研究方法正確辨識所有字元。

系統	完整標註提取準確率
(CRAFT) + 規則配對 [1]	XXX%
本研究端到端系統	XXX%
改善幅度	+XXX%

表 5-6: 本研究端到端系統與 (CRAFT) 結合規則配對方法的完整標註提取準確率比較。本研究系統在文字辨識與字串組合兩方面的綜合優勢使端到端準確率顯著提升。

配對策略	配對準確率
Ground Truth + 規則配對 [1]	XXX%
Ground Truth + 兩階段自動配對 (本研究)	XXX%
差異	XXX%

表 5-7: 使用 Ground Truth 辨識結果比較配對策略的性能。本研究的兩階段自動配對方法與基於規則的方法達到同等的配對準確率，證明自動學習配對策略的可行性。

### 5.4.3 不同標註格式適應性分析

本測試分析本研究方法對不同標註格式的適應能力。測試集包含三種主要的標註格式：上下標結構、對稱表示與平行排列。上下標結構將公差以上下標形式標註於主數值，例如主數值「1.50」上方標註「+0.1」、下方標註「-0.0」。對稱表示使用對稱公差符號，例如「 $1.5\pm0.05$ 」。平行排列將公差平行排列於主數值右側，例如「 $0.8 +0.1 -0$ 」。

測試集按標註格式分類，每種格式的標註數量分別為 XXX 個、XXX 個與 XXX 個。兩種配對方法均使用 Ground Truth 辨識結果，僅評估配對策略在不同格式上的表現。

表 5-8 展示兩種配對策略在不同標註格式上的配對準確率。結果顯示，基於規則的方法在上下標結構上達到 XXX%，在對稱表示上達到 XXX%，在平行排列上達到 XXX%，平均準確率為 XXX%。本研究的兩階段自動配對方法在三種格式上分別達到 XXX%、XXX% 與 XXX%，平均準確率為 XXX%。雖然基於規則的方法在某些格式上略優，但在其他格式上表現下降。本研究方法在各種格式上的準確率更為穩定，標準差為 XXX%，低於基於規則方法的 XXX%。

這種穩定性源於本研究方法的設計理念。基於規則的方法需要針對每種格式分析空間關係與特徵，設計對應的判斷規則。當遇到新格式時，需要重新分析與修改規則。本研究的兩階段自動配對方法透過神經網路學習語意特徵（如字元類別）與空間特徵（如相對位置、尺寸比例）的關聯，模型能從訓練資料中自動捕捉不同格式的組合模式。當遇到新格式時，模型能利用學到的特徵進行推理，無需重新設計規則。

進一步分析顯示，本研究方法在訓練時未充分涵蓋的格式變體上仍能維持合理的準確率。例如，測試集包含 XXX 個訓練時少見的格式變體，基於規則方法的準確率下降至 XXX%，而本研究方法仍維持 XXX%。這證明自動學習的配對策略具有較好的泛化能力，能適應格式的多樣性。

在實際應用中，工程圖的標註格式可能因客戶、產品或設計規範而異。基於規則的方法需要針對每種格式維護一套規則，增加系統開發與維護成本。本研究的兩階段自動配對方法透過訓練資料學習配對規則，當遇到新格式時，僅需提供訓練樣本即可適應，降低系統維護成本，提升方法的通用性。

## 5.5 錯誤案例分析與討論

雖然本研究方法在整體準確率與遮擋場景下均優於現有方法，但在某些情況下仍存在失敗案例。本節分析本研究方法的典型失敗案例，討論方法的限制與改進方向。透過誠實討論限制，為未來研究提供參考。

### 5.5.1 多組件字元的限制

本研究方法的一個已知限制是對多組件字元的處理能力不足。多組件字元指包含多個不連接組件的字元，例如字元「i」包含點與豎線兩個組件，字元「j」包含點與彎鉤兩個組件，字元「!」包含點與豎線兩個組件。本研究方法基於連接組件進行線段分組，假設同一字元的線段透過端點連接形成連通結構。對於多組件字元，不同組件之間無端點連接，導致方法將其視為多個獨立物件。

統計測試集中的字元類別分布，多組件字元佔總字元數的 XXX%，包含「i」XXX 個、「j」XXX 個、「!」XXX 個等。對於這類字元，本研究方法的辨識準確率為 XXX%，顯著低於整體字元準確率的 XXX%。失敗案例主要分為兩類：第一，點與主體部分被分開偵測，模型輸出兩個獨立字元，例如將「i」偵測為「.」與「l」；第二，點部分未被偵測，模型僅輸出主體部分，例如將「i」辨識為「l」。

圖 5.3 展示多組件字元失敗的典型案例。在案例 (a) 中，字元「i」的點與豎線在空間上分離，連接分組將其分為兩個組件。字元偵測模組將兩個組件分別偵測為獨立字元，輸出結果為「.」與「l」。在案例 (b) 中，字元「!」的點部分過小，在取樣與分組過程中未獲得足夠的代表點，導致模型未能偵測該組件，僅輸出豎線部分。

對整體準確率的影響分析顯示，多組件字元的失敗使整體字串準確率下降 XXX%。雖然多組件字元佔比較小，但在某些工程圖中可能頻繁出現，影響系統的可靠性。

改進方向包括兩個層次。短期改進可引入後處理規則，檢測多組件模式並合併。具體而言，當偵測到點狀物件與豎線狀物件在空間上接近且符合特定幾何關係時，將其合併為單一字元。此方法需要針對不同多組件字元設計規則，但能在不修改模型的情況下快速改善準確率。長期改進可在訓練時增強多組件樣本，或修改模型架構，引入組件關聯機制。例如，在字元偵測階段，不僅考慮連接組件的獨立性，也考慮組件之間的空間關係與幾何特徵，將相關組件歸類為同一字元。

### 5.5.2 其他典型失敗案例

除了多組件字元，本研究方法在以下幾種情況下也存在失敗案例。

第一類失敗案例為極度遮擋。當字元被遮擋超過 XXX% 時，即使保留線段連接關係，可見的線段特徵仍不足以判斷字元類別。例如，字元「E」的三條橫線均被線段遮擋，僅剩下豎線可見，模型可能將其誤判為「I」或「1」。在測試集中，XXX 個字串包含極度遮擋的字元，本研究方法的準確率為 XXX%，雖然優於點陣圖方法的 XXX%，但仍有改進空間。

第二類失敗案例為相似字元混淆。某些字元在視覺外觀上高度相似，例如數字「0」與字母「O」、數字「1」與字母「I」、數字「1」與符號「-」。雖然本研究方法在字串組合後引入語境修正，利用字串層級的語意資訊修正混淆，但在單字元情況下或語境不明確時仍可能誤判。例如，元件編號「R1C2」中的「1」可能被誤判為「I」，輸出結果為「RIC2」。在測試集中，XXX 個字串包含相似字元，混淆導致 XXX% 的失敗率。

第三類失敗案例為罕見字體。訓練資料涵蓋常見的工程圖字體，但某些客戶或設計規範可能使用特殊字體。當測試資料包含訓練時未充分涵蓋的字體樣式時，模型的辨識準確率下降。例如，某些字體的字元「a」採用雙層結構，與訓練資料中的單層「a」差異較大，導致辨識錯誤。在測試集中，XXX 個字串來自罕見字體，本研究方法的準確率為 XXX%。

第四類失敗案例為複雜標註格式。雖然本研究的兩階段自動配對方法能適應常見的標註格式，但對於極少數特殊格式，訓練樣本不足可能導致配對錯誤。例如，某些工程圖將尺寸標註拆分為多行排列，與訓練資料中的單行格式差異較大。在測試集中，XXX 個標註組採用特殊格式，配對準確率為 XXX%。

圖 5.4 展示這些失敗案例的視覺化範例。案例 (a) 展示極度遮擋導致的失敗，字元「E」僅剩豎線可見，被誤判為「I」。案例 (b) 展示相似字元混淆，數字「0」被誤判為字母「O」。案例 (c) 展示罕見字體導致的失敗，雙層「a」被誤判為其他字元。案例 (d) 展示複雜格式導致的配對錯誤，多行標註未被正確組合。

### 5.5.3 方法改進方向

基於上述錯誤案例的分析，本研究方法的改進方向包括短期改進與長期研究方向。

短期改進著重於在現有架構下提升準確率。第一，擴充訓練資料，特別是多組件字元、罕見字體與特殊格式的樣本。透過增加訓練樣本的多樣性，提升模型的泛化能力。第二，引入後處理規則，針對已知的失敗模式設計修正策略。例如，檢測多組件字元並合併、利用語境資訊修正相似字元混淆。第三，優化模型架構，例如增加斜率特徵的維度、調整分組策略的參數，提升模型對細節特徵的捕捉能力。

長期研究方向則探索更根本的技術改進。第一，結合大語言模型進行語意理解。當前方法主要依賴字元的視覺特徵與空間關係，未充分利用標註的語意資訊。透過引入語言模型，能在字串組合階段利用工程語意知識，提升配對準確率。例如，語言模型能理解「R10」是元件編號、「1.5±0.05」是尺寸標註，從而更準確地判斷組合關係。第二，研究端到端的向量圖理解方法。當前方法分為字元偵測、字元識別與字串組合三個獨立模組，模組之間的誤差會傳播累積。端到端方法能聯合優化所有階段，減少誤差傳播，提升整體性能。第三，探索少樣本學習技術。當遇到罕見字體或特殊格式時，收集大量訓練樣本成本較高。少樣本學習技術能利用少量樣本快速適應新場景，提升方法的實用性。

這些改進方向將在第六章的未來工作中進一步討論。透過誠實分析限制與提出改進方向，本研究為後續研究提供參考，推動工程圖自動化標註技術的進步。



## 5.6 點陣圖場景應用實驗

本研究方法專為向量格式的工程圖設計，利用參數化線段的幾何資訊與連接關係進行文字辨識。然而，在實際應用中，部分工程圖可能僅以點陣圖格式提供，例如掃描的紙本圖面、傳真文件或低解析度的影像檔案。為了評估本研究方法在點陣圖場景下的適用性，本節設計點陣圖應用實驗，探討將點陣圖轉換為向量表示後套用本研究方法的可行性與性能表現。

### 5.6.1 實驗設計

點陣圖應用實驗的核心思路是將點陣圖工程圖轉換為向量線段表示，再輸入本研究的字元偵測與識別模組。實驗流程包含三個階段：點陣圖向量化、向量線段處理與文字辨識。

在點陣圖向量化階段，採用 XXX 技術將點陣圖中的線條提取為參數化線段。此階段的輸出為一組線段的端點座標與曲線參數，格式與向量工程圖的線段表示一致。向量

化技術的選擇與參數設置將影響後續處理的準確率，但由於向量化本身不是本研究的核心貢獻，此處不詳述技術細節。

在向量線段處理階段，將向量化得到的線段輸入本研究方法的預處理模組。與直接處理向量工程圖相同，進行曲線轉直線、連接分組、斜率計算與線段取樣。然而，點陣圖向量化可能引入額外的挑戰。第一，點陣化過程會導致線條邊緣出現鋸齒與雜訊，向量化技術可能產生額外的短線段或斷裂。第二，掃描或壓縮過程可能改變線條的寬度與位置，影響端點連接關係的判斷。第三，遮擋處的線段在點陣化後合併為像素，向量化技術難以區分重疊的線段，可能導致連接分組錯誤。

在文字辨識階段，使用本研究訓練的字元偵測與識別模型進行辨識。由於模型是在向量工程圖上訓練，對點陣圖向量化產生的線段特性可能不完全適應。例如，向量化產生的線段端點位置可能存在微小偏差，斜率計算可能受到雜訊影響。這些因素均會影響辨識準確率。

## 5.6.2 實驗設置



實驗資料集採用本研究的向量工程圖測試集，將其轉換為點陣圖後進行實驗。具體而言，將 XXX 張向量工程圖轉換為 XXX DPI 的點陣圖，模擬掃描文件的品質。轉換過程採用標準的點陣化工具，確保與實際應用場景一致。

比較方法包括三種：第一，直接在向量工程圖上應用本研究方法（向量格式），作為性能上界；第二，在點陣圖上應用商業光學字元辨識（OCR）系統（谷歌光學字元辨識（Google OCR）），作為點陣圖場景的基準方法；第三，將點陣圖向量化後應用本研究方法（點陣圖向量化），驗證本研究方法在點陣圖場景下的性能。

評估指標採用字串準確率，與第 5.3 節的評估方式一致。為了分析不同場景下的表現，進一步將測試集分為被遮擋字串與未遮擋字串，分別評估準確率。

## 5.6.3 實驗結果與分析

表 5-9 展示三種方法在點陣圖場景應用實驗中的字串辨識準確率。結果顯示，在向量格式上，本研究方法達到 XXX% 的整體準確率，在被遮擋字串上達到 XXX%。在點陣圖場景下，谷歌光學字元辨識（Google OCR）達到 XXX% 的整體準確率，在被遮擋字串上達到 XXX%。將點陣圖向量化後應用本研究方法，整體準確率為 XXX%，在被遮擋

字串上為 XXX%。

結果分析顯示，點陣圖向量化後應用本研究方法的準確率低於直接處理向量格式，差距約 XXX%。這種性能下降主要源於兩個因素。第一，點陣化過程引入的雜訊與失真影響向量化品質，產生額外的短線段或斷裂，導致連接分組出現錯誤。當線段的端點位置因雜訊而偏移時，連接關係可能無法正確建立，影響同組查詢 (Same-Group Query) 的效果。第二，遮擋處的線段在點陣化後合併為像素，向量化技術難以區分重疊的線段，可能將原本屬於不同字元的線段錯誤地連接在一起，或將同一字元的線段錯誤地分離。

然而，點陣圖向量化後應用本研究方法在遮擋場景下仍優於谷歌光學字元辨識 (Google OCR)，改善幅度約 XXX%。這證明本研究方法保留線段連接關係的設計理念在點陣圖場景下仍具有一定優勢。即使向量化過程引入錯誤，同組查詢 (Same-Group Query) 仍能在部分遮擋場景下區分子元與遮擋物的線段，提升辨識準確率。

在未遮擋場景下，點陣圖向量化後應用本研究方法與谷歌光學字元辨識 (Google OCR) 的準確率相當，差距僅 XXX%。這是因為在未遮擋場景下，本研究方法的主要優勢（線段連接關係）發揮空間有限，性能主要取決於字元特徵提取與分類的能力。點陣圖向量化引入的雜訊對特徵提取造成干擾，部分抵消了方法本身的優勢。

圖 5.5 展示點陣圖向量化導致失敗的典型案例。在案例 (a) 中，點陣化引入的鋸齒導致向量化產生多個短線段，這些短線段的端點未能正確連接，使連接分組將原本屬於同一字元的筆畫分為多個組件。在案例 (b) 中，遮擋處的線段在點陣化後合併，向量化將其識別為單一線段，導致字元與遮擋物的線段被錯誤地歸類到同一組件。

#### 5.6.4 討論與未來工作

點陣圖場景應用實驗證明本研究方法在點陣圖向量化後仍能保持一定的性能，特別是在遮擋場景下優於傳統點陣圖光學字元辨識 (OCR) 方法。然而，性能下降顯示向量化過程的品質是影響最終準確率的關鍵因素。

未來研究可從兩個方向改進點陣圖場景的應用能力。第一個方向是提升向量化技術的穩健性。可針對工程圖的特性設計專門的向量化演算法，例如利用線段的方向與長度資訊過濾雜訊，或引入端點連接修正機制，自動檢測與修復斷裂的線段。此外，可設計多尺度向量化策略，在不同解析度下提取線段並融合結果，提升向量化的準確率。

第二個方向是設計容錯機制，使本研究方法能適應向量化引入的錯誤。例如，在連

配對策略	上下標結構	對稱表示 ±	平行排列	平均
規則配對 [1]	XXX%	XXX%	XXX%	XXX%
兩階段自動配對 (本研究)	XXX%	XXX%	XXX%	XXX%

表 5-8: 兩種配對策略在不同標註格式上的配對準確率比較。本研究的兩階段自動配對方法在各種格式上表現穩定，證明透過學習適應格式多樣性的有效性。

圖 5.3: 本研究方法在多組件字元上的典型失敗案例。(a) 字元「i」的點與豎線被分開偵測為兩個獨立字元；(b) 字元「！」的點部分未被偵測，僅輸出豎線部分。紅色框標示錯誤的偵測結果。



圖 5.4: 本研究方法的其他典型失敗案例。(a) 極度遮擋導致字元「E」被誤判為「I」；(b) 相似字元混淆，數字「0」被誤判為字母「O」；(c) 罕見字體導致雙層「a」辨識失敗；(d) 複雜標註格式導致配對錯誤。紅色標記指出錯誤位置。

方法	整體字串準確率	被遮擋字串準確率	未遮擋字串準確率
向量格式 + 本研究方法	XXX%	XXX%	XXX%
點陣圖 + 谷歌光學字元辨識 ( Google OCR )	XXX%	XXX%	XXX%
點陣圖向量化 + 本研究方法	XXX%	XXX%	XXX%

表 5-9: 點陣圖場景應用實驗的字串辨識準確率比較。點陣圖向量化後應用本研究方法的準確率低於直接處理向量格式，但在遮擋場景下仍優於點陣圖光學字元辨識 (OCR) 方法。

圖 5.5: 點陣圖向量化導致的典型失敗案例。(a) 點陣化鋸齒導致向量化產生斷裂，連接分組錯誤；(b) 遮擋處線段在點陣化後合併，向量化無法區分重疊線段，導致分組錯誤。紅色標記指出錯誤位置。

接分組階段，可引入容錯距離閾值，當端點距離小於閾值時仍視為連接，允許微小的位置偏差。在分組階段，可設計多假設策略，對於模糊的連接關係產生多個分組假設，透過後續的字元識別結果選擇最可能的分組。此外，可在訓練時引入向量化雜訊的模擬，使模型能學習處理斷裂與誤連接的線段，提升在點陣圖場景下的泛化能力。

雖然本研究方法在點陣圖場景下的準確率低於直接處理向量格式，但仍展現了一定的應用潛力。對於僅提供點陣圖格式的工程圖，透過向量化轉換後應用本研究方法，能在遮擋場景下提供優於傳統光學字元辨識 (OCR) 方法的性能。這為本研究方法的實際應用提供了更廣泛的場景支援。



## 第6章 結論與未來工作

本研究因應製造業智慧化的需求，針對工程圖自動化標註提出了向量文字辨識與字串組合兩大目標。為了達成兩大目標，本研究引入進階點網路 (PointNet++) 作為核心骨幹網路，並針對向量工程圖的特性進行模型架構與運作機制的調整。首先，本研究提出線段組感知最遠點取樣 (Line Group-Aware FPS) 與基於連接關係的同組查詢 (Same-Group Query)，改善了遮擋場景下特徵混雜的問題，使模型能在遮擋處獲取正確的字元結構特徵，顯著提升被遮擋字串的辨識準確率。此外，藉由引入雙角度轉換的斜率特徵編碼，消除了線段方向的模糊性，使模型能學習更穩定的特徵表示。在字串組合方面，本研究提出兩階段自動配對方法，結合語意特徵與空間特徵進行學習，能適應不同格式的標註結構，相較於基於規則的方法具有更好的泛化能力。實驗結果證明，本研究方法在遮擋場景下的字串辨識準確率顯著優於商業光學字元辨識 (OCR) 系統，在標註格式適應性上與基於規則方法達到同等性能，達成實用的工程圖自動化標註目標，有助於提升製造業的數位化與自動化水準。

雖然本研究的方法相較於先前研究有著明顯的優勢，不過，本論文方法仍有以下限制。



- **多組件字元的處理能力：**本研究方法基於連接組件進行線段分組，假設同一字元的線段透過端點連接形成連通結構。對於多組件字元 (如「i」、「j」、「!」)，不同組件之間無端點連接，導致方法將其視為多個獨立物件。在測試集中，多組件字元的辨識準確率顯著低於整體準確率，使整體字串準確率下降 XXX%。未來研究可引入後處理規則，檢測多組件模式並合併，或在訓練時增強多組件樣本。長期而言，可修改模型架構，引入組件關聯機制，考慮組件之間的空間關係與幾何特徵，將相關組件歸類為同一字元。
- **相似字元的混淆：**某些字元在視覺外觀上高度相似，例如數字「0」與字母「O」、數字「1」與字母「I」、數字「1」與符號「-」。雖然本研究方法在字串組合後引入語境修正，但在單字元情況下或語境不明確時仍可能誤判。在測試集中，相似字元混淆導致 XXX% 的失敗率。本研究認為這是字元級別辨識方法的通用問題，需要在字串或標註層級引入更強的語意約束。未來研究可引入基於變換器 (Transformer) 的序列建模，將字串視為字元序列，透過雙向上下文資訊提升辨識準確率，或結合工程領域的語言模型，利用工程術語的語意知識修正混淆。
- **點陣圖場景的應用能力：**本研究方法專為向量格式的工程圖設計，直接處理參數

化線段的幾何資訊與連接關係。然而，在實際應用中，部分工程圖可能僅以點陣圖格式提供，例如掃描的紙本圖面或低解析度的影像檔案。為了擴展方法的適用範圍，未來研究可探索將點陣圖轉換為向量表示後套用本研究方法的可行性。具體而言，可採用向量化技術將點陣圖中的線段提取為參數化表示，再輸入本研究的字元偵測與識別模組。然而，點陣化過程會引入雜訊與失真，且向量化技術可能無法完美重建線段的端點連接關係，導致連接分組出現錯誤。此外，點陣圖的解析度、掃描品質與壓縮方式均會影響向量化結果。初步實驗顯示，在點陣圖場景下，本研究方法的準確率低於直接處理向量格式，但仍優於傳統點陣圖光學字元辨識 (OCR) 方法。未來研究可針對點陣圖向量化的特性，設計更穩健的預處理流程與容錯機制，或開發專門的點陣圖辨識模組，結合向量方法的優勢與點陣圖方法的適應性，提升方法在混合格式場景下的性能。

綜上所述，本研究為工程圖自動化標註提供了一套有效的解決方案，在遮擋場景處理與格式適應性方面取得顯著進展。透過保留向量資料的幾何資訊與連接關係，克服了點陣圖方法的固有限制。透過自動學習配對規則，降低了系統開發與維護成本。實驗結果證明方法的有效性與實用性。未來研究將聚焦於多組件字元處理、語意理解增強與點陣圖場景適應，進一步提升方法的準確率與適用範圍，推動製造業智慧化的發展。

## 參 考 文 獻

- [1] J. Villena Toro, A. Wiberg, and M. Tarkian, “Optical character recognition on engineering drawings to achieve automation in production quality control,” *Frontiers in manufacturing technology*, vol. 3, p. 1154132, 2023.
- [2] J. V. Toro and M. Tarkian, “Optimizing text recognition in mechanical drawings: A comprehensive approach,” *Machines*, vol. 13, no. 3, p. 254, 2025.
- [3] T. Y. Zhang and C. Y. Suen, “A fast parallel algorithm for thinning digital patterns,” *Communications of the ACM*, vol. 27, no. 3, pp. 236–239, 1984.
- [4] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, “Character region awareness for text detection,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9365–9374, 2019.
- [5] X. Jiang, L. Liu, C. Shan, Y. Shen, X. Dong, and D. Li, “Recognizing vector graphics without rasterization,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 24569–24580, 2021.
- [6] W. Liu, T. Yang, Y. Wang, Q. Yu, and L. Zhang, “Symbol as points: Panoptic symbol spotting via point-based representation,” *arXiv preprint arXiv:2401.10556*, 2024.
- [7] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, “Pointnet: Deep learning on point sets for 3d classification and segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017.