

智慧誘捕籠：剪耳辨識

朝陽科技大學 A00000000 ○○○

朝陽科技大學 A00000000 ○○○

朝陽科技大學 A00000000 ○○○

朝陽科技大學

A00000000 ○○○

摘要

本專題要做的是：用影像辨識判斷一隻流浪貓有沒有被剪過耳朵。TNR(捕捉—絕育—回置)會把已經結紮的貓剪掉一邊的耳尖當作記號，所以「耳尖有沒有被剪掉」是現場唯一看得出來的線索。這件事對辨識來說有三個難處：要分的兩類是同一種動物的兩種狀態，差別只在耳朵尖端那幾公分；剪過耳朵的照片天生就比沒剪的少很多；辨識還要在一台運算能力很小的裝置上當場跑完。本團隊已經做出原型並放到現場運作，但判斷成功的次數仍然不夠。本專題打算用三個方法改善——兩階段的遷移學習、訓練時的資料增強，以及把「看一張就下結論」改成「累積好幾張再下結論」。目的不是追求某個準確率數字，而是驗證這三個方法到底有沒有用；就算結果是沒有用，也要能說清楚卡在哪裡。

1. 前言



同一隻貓：左耳完整、右耳已剪耳 (TNR 僅剪除單側耳尖)

圖 1. 剪耳長什麼樣。同一隻貓一定是一邊完整、一邊被剪平；兩類的差別只在耳朵尖端那幾公分，這就是本專題要自動認出來的東西。

研究動機。臺灣從 2017 年停止用安樂死控制族群數量之後，收容所的容量就成了主要限制 [1]，降低繁殖率因此變成主要的管理方式。TNR 是目前文獻裡研究最完整、對族群數量影響最明確的做法 [2]，能不能成功，關鍵在於抓不抓得到還沒結紮的貓；而現場唯一看得出來的線索，就是耳尖有沒有被剪掉 (圖 1) [3]。傳統誘捕籠不會分辨這個記號，只要走進籠子就一律關門；在長期做 TNR 的地區，已結紮的比例逐年累積，多數捕獲其實都是重複抓到同一批貓。如果籠子能在門口就自動分辨剪耳，就只會抓到真正需要處理的個體。

裝置條件。本團隊的原型是在市售誘捕籠上加裝一個控制盒，裡面有相機和一顆微控制器等級的運算單元。辨識要在這台裝置上當場跑完，不會連到伺服器算。這一點框住了本專題能用的方法：模型必須先壓縮成裝置跑得動的格式，判斷流程也要能用現有韌體寫出來。在電腦上測起來再好、但裝不進這台裝置的做法都不列入考慮。

這件事難在哪。這個任務跟一般的物件偵測不一樣，難處有三個。第一，要分的差別太細：兩類都是貓，差別只在耳朵尖端那幾公分的形狀 (圖 1)，不像「貓和狗」那樣一眼就分得出來。第二，兩類的照片數量差很多：TNR 只剪一邊的耳尖，所以同一張照片裡一定是一隻剪過、一隻沒剪，剪耳的樣本天生就比較少——偏偏剪耳才是判斷有沒有結紮的唯一依據。第三，看單一張照片不可靠：貓在籠子裡的角度、距離、被擋住的程度一直在變，同一次進籠拍到的照片，多數都不足以下判斷 (圖 2)。

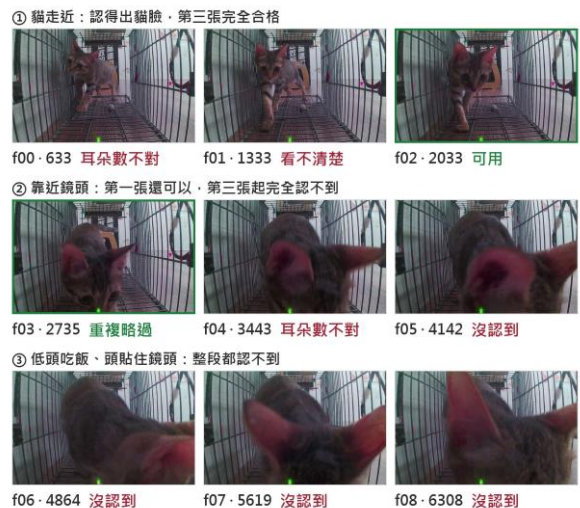


圖 2. 同一隻貓一次進籠拍到的九張照片。角度、距離和遮擋一直在變，單張的結果很不穩定——九張裡只有一張夠格當判斷依據，這就是改成累積多張的理由。

別人做到哪裡。現有的智慧誘捕籠研究，處理的是「分辨是不是目標物種」[5]，那種任務兩類長得很不一樣，跟我們要分的細微差別不同。至於辨識要放在裝置上還是雲端，已經有比較延遲和耗電的研究 [4]，但那些研究多半假設辨識結果可以直接拿來用，沒有處理「這張照片根本不夠判斷」的情況。對我們來說，

只把準確度拉高不夠，還要讓系統自己看得出來哪一張不能用。

本專題做什麼、不做什麼。硬體、韌體和開關門的機構都已經做好並在現場運作，不在本專題範圍內。本專題只處理辨識方法本身：模型怎麼訓練、判斷條件怎麼設計，以及怎麼驗證這些做法有沒有效。全部都用現有的裝置和現有的資料改良，不加新硬體。整體流程見圖 3。

2. 方法

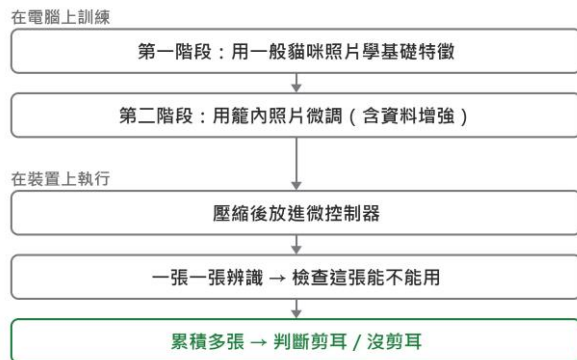


圖 3. 方法流程。上半是在電腦上做的兩階段訓練，下半是壓縮後放進裝置、一張一張辨識再累積起來下判斷。

2.1. 兩階段遷移學習。遷移學習就是先拿別人在大量影像上訓練好的模型當起點，再改成自己要的任務。本專題分兩階段：先用一般貓咪照片，讓模型學會「貓臉」和「耳朵」長什麼樣；再用籠內實際拍到的照片做微調，讓它適應籠內的角度、光線和距離。只用籠內照片訓練，數量太少學不穩；只用一般照片又沒有籠內條件，所以兩種都要用，而且要分先後。

2.2. 訓練時的資料增強。資料增強就是在訓練時把同一張照片做一些變化(轉角度、調亮暗、縮放)，讓模型看到更多種情況，用來補剪耳照片太少的問題。要注意的是，增強不一定有效，所以本專題會比較「有增強」和「沒增強」兩種訓練結果，分別看剪耳這一類的表現差多少，用實驗來確認，不用猜的。

2.3. 累積多張照片再下判斷。現在的做法是等觀察時間結束，直接比對偵測到的框數，等於讓單張照片決定結果。本專題改成：每一張照片先過一輪合理性檢查，通過的才算數；沒通過的就標成「這張不能用」並記下原因，而不是直接丟進去算。這樣分開之後，系統就分得出「照片不夠判斷」和「照片夠但確實沒看到剪耳」——前者應該什麼都不做，後者才是真的判斷結果。

2.4. 壓縮後放進裝置。改良後的模型必須先量化(把模型壓縮成裝置跑得動的格式)，才能放到運算能力很小的微控制器上即時辨識。壓縮會讓準確度掉一些，所以本專題會比較壓縮前後每一類的表現，確認判斷要用的那一類沒有被壓掉。

2.5. 怎麼驗證這些方法有沒有用。每一張照片的辨識結果、以及它過了哪一關沒過哪一關，都會記下來。有了這些紀錄就能算出三件事：可用照片佔多少比例、每一關各擋掉多少張、以及每一類各抓到幾成(逐類別召回率)。第二項最關鍵：照片如果多半被「信心不夠」擋掉，代表問題在模型；被「畫面不合理」擋掉，代表問題在拍攝條件。兩者要改的方向完全不同，這也是我們堅持保留逐張紀錄的原因。

2.6. 為什麼不看「整體正確率」。測試場地大約八成的貓已經結紮，所以只要程式一律回答「已結紮」，整體正確率就有八成——這個數字看起來漂亮，其實什麼都沒說明。本專題一律分開看每一類的表現，再加上各類之間互相認錯的情形。

3. 預期結果

項目	現在的做法	本專題要驗證的做法
判斷依據	單張照片直接比數量	累積多張再下判斷
不能用的照片訓練方式	分不出來 一次訓練完	標示出來並記下原因 分兩階段(遷移學習)
兩類數量差很多	沒有處理	訓練時做資料增強

表 1. 現在的做法與本專題要驗證的做法對照。

本專題最後會交出三樣東西：可用照片比例提升多少、每一關各擋掉多少張，以及剪耳這一類在改良前後差多少。要先說清楚的是，結果也可能是「這些方法沒有用」——如果照片數量的限制沒辦法靠演算法補起來，提升幅度就會很有限。即使如此，那些紀錄仍然可以指出卡點是在模型還是在照片本身。能把這兩件事分開並量化出來，就是本專題的貢獻：我們要驗證的是方法不可行，而不是追求某個準確率數字。

參考文獻

- [1] Yan, T.-Y., and Teng, K. T.-y. Trends in animal shelter management, adoption, and animal death in Taiwan from 2012 to 2020. *Animals*, 13:1451, 2023.
- [2] Smith, L. M., Hartmann, S., Munteanu, A. M., Dalla Villa, P., Quinnell, R. J., and Collins, L. M. The effectiveness of dog population management: A systematic review. *Animals*, 9:1020, 2019.
- [3] Agriculture Knowledge Portal, Taiwan. Introduction to TNR. 2020.
- [4] Koubaa, A., Ammar, A., Alahdab, M., Kanhouh, A., and Azar, A. T. DeepBrain: Experimental evaluation of cloud-based computation offloading and edge computing in the Internet-of-Drones for deep learning applications. *Sensors*, 20:5240, 2020.
- [5] Besala, F. I., Niimoto, R., Lee, J. H., and Okamoto, S. Development of AI-based smart box trap system for capturing a harmful wild boar. *ROBOMECH Journal*, 12:4, 2025.