Coarse to Refine + 3D Voxel

出自

題目

Gen6D: Generalizable Model-Free 6-DoF Object Pose Estimation from RGB Images

1. Generalizable : 就是 Novel Pose Estimation

Introduction

背景

動機

1. 大多數方法需要 3d 模型
2. 大多數方法需要 category specific

目的

reference images 是多張 posed image ⇒ 要知道 image 中該物件的 pose，這樣才能算出他的座標空間

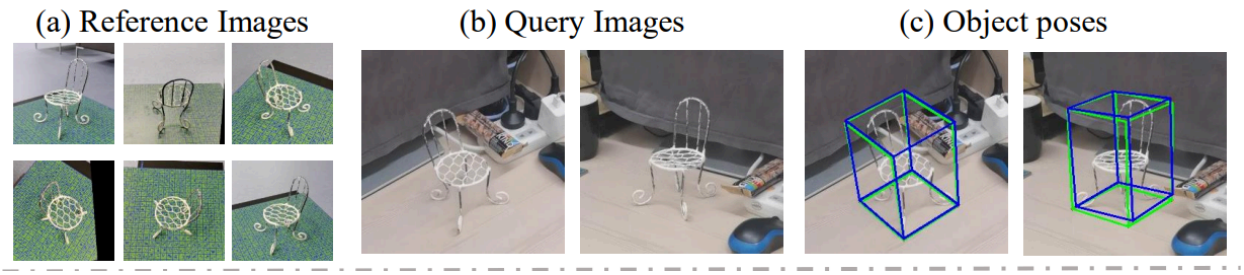


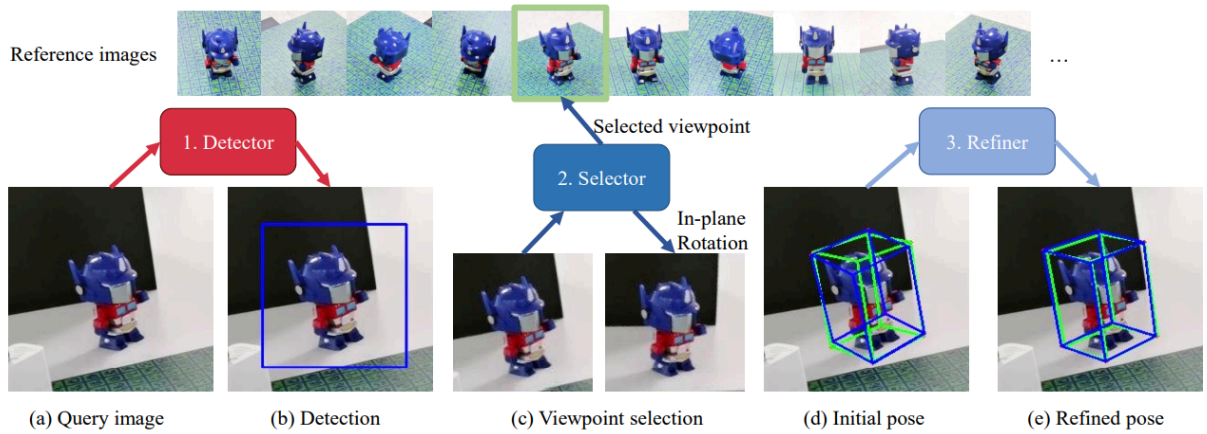
image-matching based framework

Gen6D = object detector + viewpoint selector + pose refiner

object detector : 依照 refference image 去偵測 object

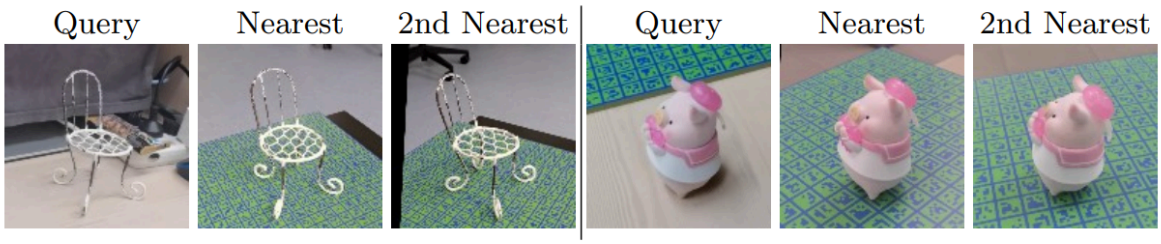
viewpoint selector： matches 最靠近 query image 的 reference image 而得到 initial pose

pose refiner : refine the pose (coarse-to-fine)



困難點 :

1. Images are sparse : refference images 不能完整覆蓋物體的所有角度或細節。
2. cluttered background : 背景干擾，導致 embed images into feature vectors 時效果不好而造成 initial pose 有可能不小心選到比較不像的



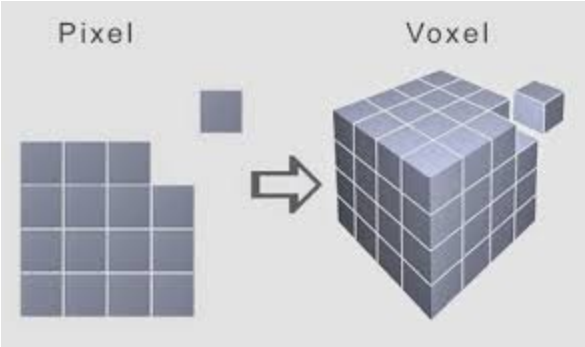
解法

1. pixel-wise comparison ⇒ concentrate on object regions 而不會被背景干擾
2. 用 global normalization layers and self-attention layers to share similarity information across different reference images. reference images to commute with each other【問題 : 是怎麼用的】

本篇提出的 3D volume-based pose refinement method

1. 找幾張與 initial pose 接近的 reference image
2. 先將 reference images 經過 CNN 變成 feature map H * W * C，接著這些 H * W 對應回 3D
3. 我們會定義一個固定大小的 voxel (三維中的 Pixel， 是很多格子)，並將剛剛那些 3d feature 依照座標位置放到 voxel 中。如果 voxel 中有不只一個 feature 則經過

pooling 得到唯一的 feature，如果 voxel 中沒有 feature 則該 voxel feature 為 0 。這個 feature 即是 feature volume



4. 接著我們一樣製作一個 Query image 的 feature volume， 並將 reference image feature volume 與 query image feature volume 丟給 3d CNN refine the pose

好處 :

1. 不用 render new image
2. refine 時是用 3D 資訊，而不是用 2d to regress 3d

相關研究

model-free : 有些模型會在 path 中重建 3d model，因此不需要額外 3d model

Category-specific : 可以偵測同 category 的物體

實作 Generalizable pose estimators 的方法 :

1. shape embedding
2. template matching
3. rendering and-comparison : 將 3D Model 依照預測的 input pose render，而不是像 template matching 一樣生成一堆沒用的 template。 現存的 refiner 都用

render and comparison 來 refine the predict

4. neural rendering techniques : render from posed images， 可以達到 model free 但是如果 appearance 改變例如光照效果就會很差。 因此有些人會加 additional depth maps [41] or object masks [41,73]

Instance detection : given a reference image 找出物體

有一些 instance detection 除了可以偵測到 instance 之外，也會順便預測 instance 的 pose

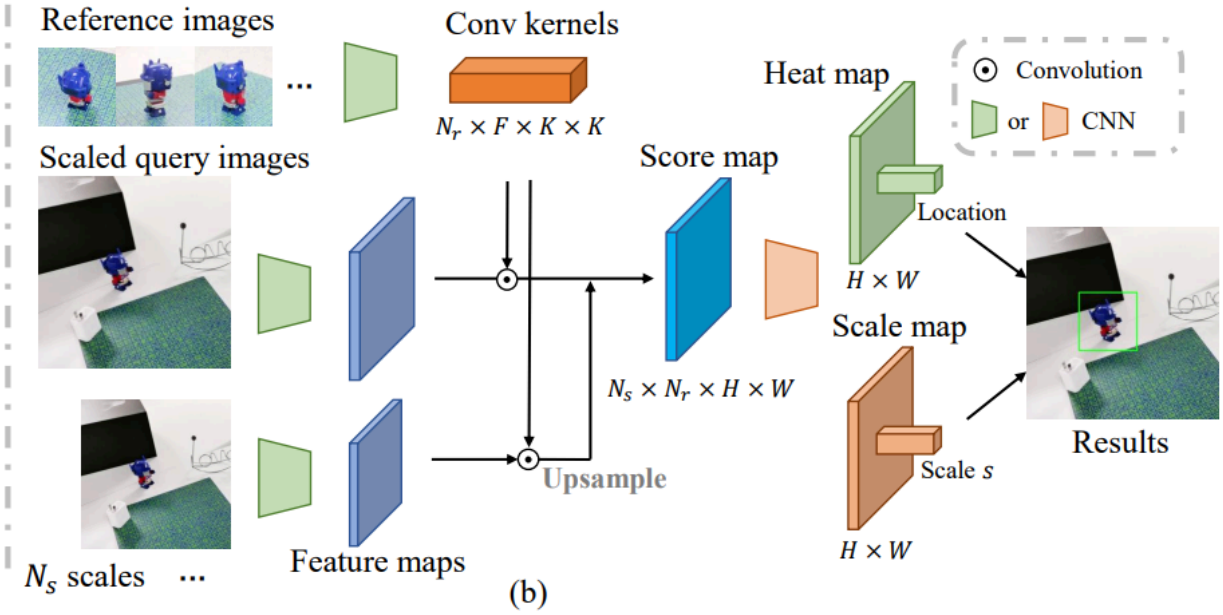
方法

1. **世界坐標系 (World Coordinate System) :** 真實場景中的全局坐標系
2. 相機坐標系 (Camera Coordinate System) : 物體相對於相機的位置與姿態
3. 物體坐標系 : 先透過 epipolar line 與 找到對應點 (triangulating points) 來估算物體的 size，接著將物體中心放到原點，並且單位長設為 size ，也就是說當前物體是在單位圓內。這個坐標系的目的是希望能夠製作 feature volume

correlation-based instance detector :

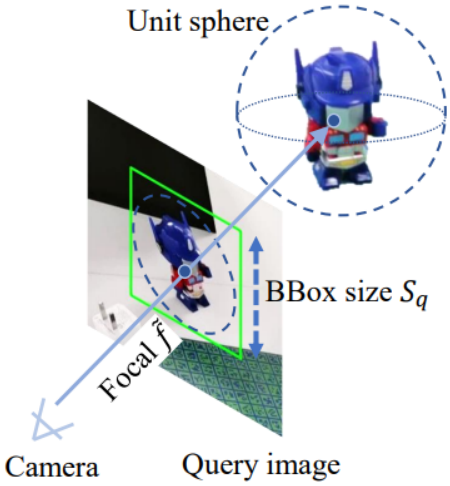
第一步，利用 query image 與 reference image 以 image matching 的方法得出 heat map 與 scale map ， 得到 object center 與 scale

而對於 scale differences ，本篇論文透過 resizing the query images to different scales 來解決這個問題



接著透過預測到的 object center 與 s 來推估出 object 的深度與相機座標，需要先得知相機參數 f ，如果有換相機，需要用新的 f

$$d = 2\tilde{f}/S_q$$



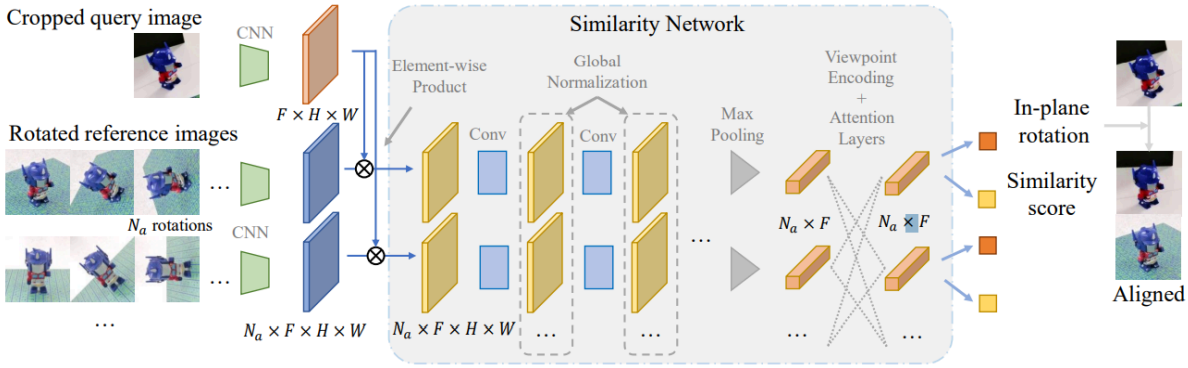
(a)

Viewpoint selection : 選一張 reference image 並且判斷 in-plane rotation

把每一張 reference image 做 in-plane rotation，沿著光軸旋轉，也就是整個照片旋轉 (順帶一提， out-of-plane rotation 會將飛機模型仰角提高以及改變環視角度，看到不同機翼)

於是我們有 Nr * Na 張 reference image，經過 cnn 生成 Nr * Na * F 個 feature map ，每一個 feature map 與 query feature map 做 element-wise product 生成 correlation score map。接著會經過兩道 CNN 並且經過 CNN 的 Feature Map 要去做 Global Normalization ，計算每一個 feature map 的 mean 與 varience 然後 normalization ， 最後經過 max pooling 後丟入 transformer ， input sequence 長度 Na * Nr ，每個向量長 F 彼此作 attention 最後生成 in-plane rotation 與 similarity score ，選擇 score 最高的那個得到 rotation 的 initial prediction

transformer 有做 positional encoding



3D volume-based pose refiner

將 [-1, 1] ^ 3 的單位立方體分割成 32 * 32 * 32 個 voxel

選 Nn 張最近的 reference image， 他們各自經過 2D CNN 可以變成 Nn 張 feature map，feature map 的每一個 pixel 是一個 activation，是一個 c 維向量。接著會依照 2D-3D 關係對應到 voxel 中，所以每個 voxel 中有 k 的 c 維向量，接著我們會以 r_mean 與 r_var 來代表該 voxel。

query 也可以放到 voxel 中，得到 q_mean 與 q_var， 接著我們得到 f_mean = concate(q_mean, r_mean), f_var = concate(q_var, r_var) 作為 voxel 的代表

將 voxel 丟入 3D CNN 中預測 pose residual， 一步一步將 pose 變得精準

📌 3D CNN 流程

1. **輸入體積 (Input Volume)**

```yaml
Input: (32, 32, 32, 1)   # 最後一個 1 是 channel
```

2. **第一層 3D 卷積 (Conv3D)**
   - kernel: 3 × 3 × 3
   - filters: 32
   - stride: 1, padding: same
   - output shape: (32, 32, 32, 32)

residual = 2D in-plane offset, a scale factor and a residual 3D rotation.

從而避免直接預測 translation 而讓 object 離開單位球

會重複 refine 三次

Experiments

model-free datasets: the MOPED dataset and a new GenMOP dataset.

可以學習的地方

問題