國立臺灣大學電機資訊學院電子工程學研究所

碩士論文

Graduate Institute of Electronics Engineering
College of Electrical Engineering and Computer Science
National Taiwan University

Master's Thesis

基於物件偵測之多平面物體姿態估測系統

Multiple Planar Object Pose Estimation System Based on Object Detection

劉家甫

Chia-Fu Liu

指導教授：簡韶逸 博士

Advisor: Shao-Yi Chien, Ph.D.

中華民國 113 年 01 月

January 2024

doi:10.6342/NTU202400430

國立臺灣大學碩士學位論文
口試委員會審定書
MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

基於物件偵測之多平面物體姿態估測系統

Multiple Planar Object Pose Estimation System Based on Object Detection

本論文係 劉家甫 (姓名) R08943134 （學號）在國立臺灣大學電子工程學研究所完成之碩士學位論文，於民國 113 年 01 月 30 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Graduate Institute of Electronics Engineering on 30 (date) January (month) 2024 (year) have examined a Master's Thesis entitled above presented by Chia-Fu Liu (name) R08943134 (student ID) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

簡仁宗　施吉昇　黃文皇

（指導教授 Advisor）

陳冠文

系（所、學位學程）主管 Director: 江介宏

致謝

光陰荏苒，碩士生涯也終於在此告一個段落，回想一路上的顛簸，包括研究的不順利與更換指導教授等，現在能完成這篇碩士論文，心中充滿了無限的感激。

首先要感謝的是我的指導教授簡韶逸老師，在我陷入最低潮的時候讓我加入了實驗室，老師的溫暖彷彿一道穿過濃密烏雲的陽光，讓我在黑暗中又看到了一絲希望能夠繼續走下去。在研究的過程中，老師也給予學生很大的自由，在不施加過大的壓力下卻又能適時地提點方向給予寶貴的建議，真的非常感謝老師這幾年的照顧與指導，也祝福老師的公司和實驗室都能蓬勃發展。

第二位是開設 HLS 課程的賴瑾老師，在擔任老師助教的期間，接觸了許多不一樣的領域，學會了很多從來沒想過會去碰的東西，也從老師豐富的經驗中學習到了許多的軟實力，真的獲益良多，能夠與老師共事並參與基金會的草創時期是我的榮幸。

第三位是實驗室的同學華揚，從我進入實驗室就開始帶我，並給予我各項參考資料，甚至在畢業後仍然持續跟我討論提供建議與幫忙，如果沒有你的協助我大概也無法順利完成這篇論文，真的非常感謝你。

然後是實驗室的博班學長致廷、柏憲、立洋，同屆的汶璁、在賢，學弟妹紹軒、昱愷、凱翔、子傑、可瀚等等人數實在太多無法一一列出，還有一起口試的昱仁、奕勳、愷縴，因為有實驗室的大家，碩士生涯的後半段才能過得如此快樂，在加入這裡以前從沒想過實驗室的氛圍可以這麼愉快，卻同時又充滿了各種強者，感謝大家這幾年的陪伴，我不會忘記與大家一起討論研究、吃飯出遊，和天南地北亂聊的歡樂時光。

我也要感謝大學到現在的好朋友騰勻、玉麟、郁閑，從大學時期就在各個活動中建立起革命情感，一直無話不聊到現在，每次回到台南跟你們見面聊天都能夠讓我充電再繼續前進，雖然大家現在工作遍布各地，但我們的友情絕對不會斷，感謝有你們讓我的人生更豐富多彩。

doi:10.6342/NTU202400430

而這一路上，最要感謝的就是我的家人以及女友，從小到大外婆就一直寵愛著我，媽媽總是不間段的關心照顧與叮嚀我，爸爸和弟弟時常與我討論研究與技術上的問題，女友家稜一直在我身邊給予我鼓勵、照顧與陪伴，真的非常感謝你們讓我能夠沒有任何後顧之憂的一路走到現在，我想我真的非常幸福。

最後，我也要感謝自己，在數度想要放棄的時候選擇堅持下來，走完這條充滿荊棘的道路。

僅將本篇論文獻給在我人生中幫助過我的所有人們。

2024.01.31 劉家甫

doi:10.6342/NTU202400430

中文摘要

隨著終端裝置技術的快速發展，擴增實境（AR）和混合實境（MR）正受到越來越多的關注。估測平面物體六自由度（6-DoF）姿態是這些應用的關鍵之一。多年來，人們進行了大量研究以估測單一已知平面目標的姿態。然而，在實際應用中可能會遇到多個平面目標，並且根據具體應用可能會替換為不同的目標。我們的目標是設計一個通用的平面物體姿態估測系統，能夠在最小的訓練要求下高效地估測多個任意平面物體的姿態。

在這篇論文中，我們提出了一個基於物件偵測和直接姿態估測算法的多平面物體姿態估測系統，稱為 DetDPE。我們只使用少量合成影像數據對現成的物件偵測器進行微調。物件偵測器識別用於姿態估測的平面物體，並顯著降低估測算法的複雜度。我們的 DetDPE 系統展現了高效性，同時保持了原始直接姿態估測算法的高精準度。此外，我們將物件偵測方法與基於特徵的姿態估測算法結合。結果顯示這種方法確實可以提升基於特徵的算法的性能。因此，我們可以將此物件偵測方法視為一個可應用於各種姿態估測算法的框架。

doi:10.6342/NTU202400430

Multiple Planar Object Pose Estimation System Based on Object Detection

Chia-Fu Liu

Advisor: Shao-Yi Chien

Graduate Institute of Electronics Engineering

National Taiwan University

Taipei, Taiwan

January 2024

doi:10.6342/NTU202400430

Abstract

With the rapid growth of technologies on edge devices, Augmented Reality (AR) and Mixed Reality (MR) are gaining more and more attention. Estimating the six degrees of freedom (6-DoF) planar object pose is one of the keys to these applications. Throughout the years, numerous research studies have been conducted to estimate the pose of a single known planar target. However, in real-world applications, the presence of multiple planar targets may be encountered, and they might be replaced with different ones depending on the specific application. Our objective is to design a general planar object pose estimation system capable of estimating the poses of multiple arbitrary planar objects efficiently with minimal training requirements.

In this thesis, we propose a multiple planar object pose estimation system, DetDPE, based on object detection and a direct-based pose estimation algorithm. We fine-tune an off-the-shelf object detector with only a modest amount of synthetic image data. The object detector identifies the planar objects for pose estimation and significantly reduces the complexity of the estimation algorithm. Our DetDPE system demonstrates efficiency while maintaining the high accuracy of the original direct-based algorithm. Furthermore, we integrate the object detection approach with a feature-based pose estimation algorithm. The results show that this approach can indeed enhance the performance of feature-based algorithms. Therefore, we can regard the object detection approach as a framework applicable to various types of pose estimation algorithms.

doi:10.6342/NTU202400430

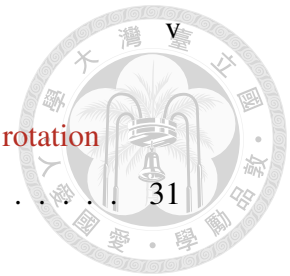
Contents

doi:10.6342/NTU202400430

doi:10.6342/NTU202400430

List of Figures

doi:10.6342/NTU202400430



doi:10.6342/NTU202400430

List of Tables

doi:10.6342/NTU202400430

doi:10.6342/NTU202400430

Chapter 1

Introduction

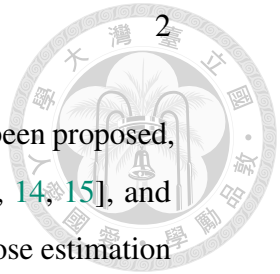
Estimating the six degrees of freedom (6-DoF) object pose is a classical problem in computer vision that seeks to determine the orientation and position relationship between a target object and a calibrated camera. It is also the key to many applications, including robotics [5], Augmented Reality (AR) [6], and Mixed Reality (MR) [7]. Despite the extensive body of research in this domain, the rapid and accurate estimation of poses for planar targets, particularly multiple arbitrary ones, remains a persistently formidable task. Our goal is to design a general planar object pose estimation system capable of estimating the poses of multiple arbitrary planar targets efficiently with minimal training requirements.

1.1 6-DoF Planar Object Pose Estimation

The pose of a rigid object is characterized by three rotation and three translation components, constituting a total of six degrees of freedom. When the pose determination process relies solely on image data, it is termed *pose estimation*. Conversely, *pose tracking* involves ascertaining the object's poses within a sequential series of camera frames by leveraging the known poses from preceding frames. In this thesis, we only focus on the pose estimation of planar objects.

Fiducial markers are well-designed patterns with distinctive and robust visual

doi:10.6342/NTU202400430



characteristics. Throughout the years, various types of markers have been proposed, including squared markers [8, 9, 10, 11, 12], circular markers [13, 14, 15], and learned markers [16, 17]. Given their easily distinguishable nature, pose estimation based on these patterns demonstrates both efficiency and accuracy. Nonetheless, their applications are limited by the necessity of using specific patterns.

As deep learning-based methods have proven to be effective in numerous computer vision tasks, several end-to-end frameworks for estimating 3D objects' poses have been devised [18, 19]. However, most approaches for planar objects are still built upon traditional modular pipelines, primarily attributed to the scarcity of available training data.

Existing pose estimation algorithms for arbitrary planar targets can be broadly classified into two categories: *feature-based approaches* and *direct approaches*.

Feature-based approaches begin by extracting features from both the planar target and the camera frame. The central idea is to establish a set of correspondences between the 3D points of the target object and their 2D projections. This correspondence set serves as the basis for estimating the pose relationship between the target and the camera. Over the past few decades, several feature extraction methods have been developed [20, 21, 22, 23, 24]. To enhance the robustness of the matches, various RANSAC [25, 26] techniques have been employed to eliminate outliers. Subsequently, the final pose is computed using Perspective-n-Point (PnP) [27, 28] algorithms. Since the performance of feature-based methods relies on the successful extraction and precise matching of features between the planar target and the camera frame, their efficacy tends to diminish when dealing with textureless targets or blurry frames.

Direct Approaches, on the contrary, do not depend on features. Instead, these methods seek to determine the optimal pose from pre-defined candidates by minimizing the appearance error between the target and its projection onto the camera frame. For planar objects, solving the 6-DoF pose estimation problem can be simplified to 2D template matching, involving both iterative [29, 30, 31, 32, 33,

doi:10.6342/NTU202400430

34, 35] and non-iterative [36, 37, 38, 2] approaches.

Despite the considerable efforts invested in this domain, to the best of our knowledge, there is a notable gap as no existing work specifically concentrates on the pose estimation of multiple planar targets.

1.2 Object Detection

The primary challenge in multiple planar object pose estimation lies in efficiently matching objects present in the camera frame with targets stored in the database. All the single-target approaches mentioned in Section 1.1 operate under the assumption that the target to be matched is already known. Simply matching the frame with all the targets in the database is highly inefficient and may have a significant number of outliers. With the rapid growth in the field of object detection, we can leverage object detectors to identify our targets and further reduce the complexity of traditional methods.

In recent years, deep learning methods have dominated the field of object detection. We can classify these methods into three main groups: two-stage [39, 40, 41], one-stage [42, 43, 44, 45, 4], and transformer-based [46, 47]. The two-stage object detection networks first generate region proposals, indicating the locations of objects. The regions are then classified into the categories they belong to. In contrast, one-stage object detection networks locate and categorize objects simultaneously through the use of Deep Convolutional Neural Networks (DCNNs), eliminating the need for a distinct partitioning into two stages. Building on the success of transformers in Natural Language Processing (NLP), researchers have also integrated transformers into object detection, yielding promising results. While two-stage and transformer-based methods exhibit commendable accuracy, their speed remains a limitation for real-time applications. Notably, significant progress in one-stage methods has enabled the attainment of both high accuracy and real-time processing speed.

doi:10.6342/NTU202400430

1.3 Contributions

Given the scarcity of planar training data and the goal of ensuring the system's generalizability—meaning there is no need to retrain a huge network when altering the target database—our approach aims to minimize training efforts while still harnessing the power of deep learning. In this thesis, we propose a multiple planar object pose estimation system named DetDPE, which integrates an object detection network with the Direct Pose Estimation (DPE) [2]. It is important to note that we exclusively utilize a limited amount of synthetic data to fine-tune a small off-the-shelf detection network. By leveraging the capabilities of the object detection network, our approach not only identifies the targets for pose estimation but also substantially reduces the complexity of the DPE process, all while maintaining a high level of accuracy. Besides, DetDPE outperforms DPE in multiple target scenarios and is more robust against occlusion. Furthermore, We evaluate this approach with a feature-based pose estimation method. The experimental results demonstrate that the object detector can enhance the performance of feature-based methods as well. To sum up, the contributions of this thesis are as follows:

- We propose a multiple planar object pose estimation system called DetDPE based on object detection and DPE, which can identify the targets and accelerate the DPE process by 9-16$\times$ while maintaining high accuracy and being more robust against occlusion.
- We show that it is adequate to exclusively employ a modest amount of synthetic data to train an off-the-shelf compact object detection network for detecting planar targets.
- We also assess this detection-based approach in conjunction with feature-based pose estimation. The results demonstrate its capability to enhance performance in this context as well. Consequently, we can regard it as a versatile framework applicable to various types of pose estimation algorithms.

doi:10.6342/NTU202400430

1.4 Thesis Organization

We have introduced the 6-DoF multiple planar object pose estimation and outlined our primary contributions in this chapter. Further background knowledge and related works are discussed in Chapter 2. The details of our proposed DetDPE system are presented in Chapter 3, and experimental results are provided in Chapter 4. The conclusion is finally presented in Chapter 5.

doi:10.6342/NTU202400430

Chapter 2

Background Knowledge and Related Work

In this chapter, we first introduce the mathematical formulation of the 6-DoF planar object pose estimation problem. Subsequently, some related works are presented.

2.1 Problem Formulation

For a camera frame $\mathcal{I}_c$ and a planar target image $\mathcal{I}_t$, the 6-DoF planar object pose estimation task is to determine the pose $\mathbf{p} \equiv (\mathbf{R}, \mathbf{t})$ of the planar object based on its orientation and position relative to the calibrated camera, where

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{33} & r_{32} & r_{33} \end{bmatrix} \in SO(3), \quad \mathbf{t} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} \in \mathbb{R}^3, \tag{2.1}$$

are the rotation matrix and translation vector, respectively. The transformation between the 3D points $\mathbf{x}_i = [x_i, y_i, 0]^\top, i = 1, \ldots, n, n \geq 3$ in object coordinate

doi:10.6342/NTU202400430

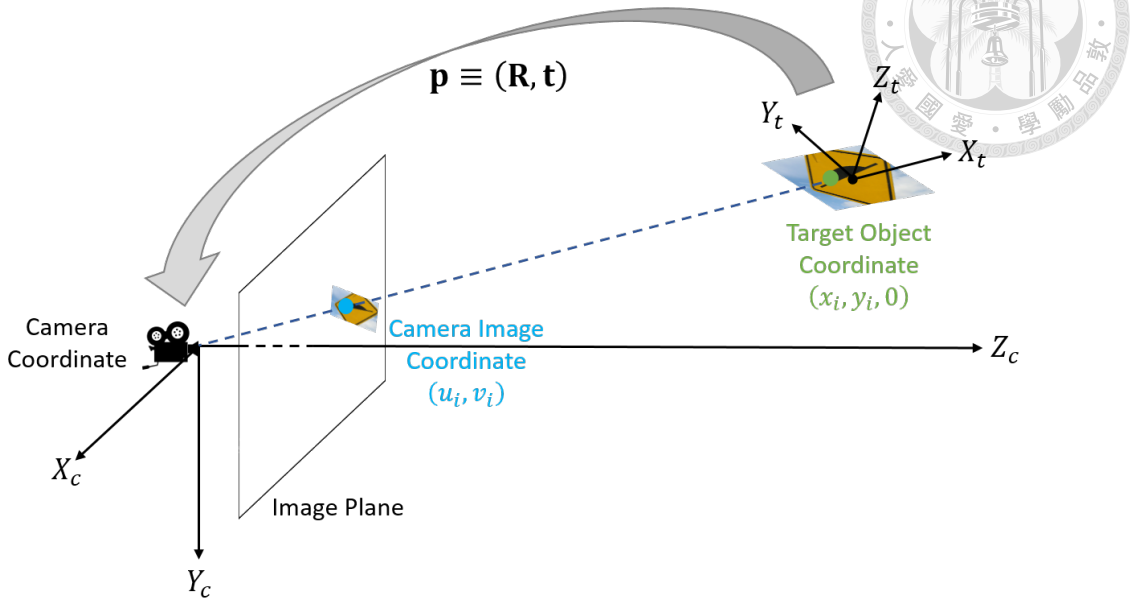


Figure 2.1: The coordinate system transformation between the planar target and the camera image.

and the camera image points $\mathbf{u}_i = [u_i, v_i]^\top$ in $\mathcal{I}_c$, can be formulated as

$$z_{ci}\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \mathbf{K}\mathbf{T}_{ext}\begin{bmatrix} x_i \\ y_i \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} \mathbf{R}|\mathbf{t} \end{bmatrix}\begin{bmatrix} x_i \\ y_i \\ 0 \\ 1 \end{bmatrix} \tag{2.2}$$

where z_{ci} is the depth value of the 3D point in the camera coordinate system, and $\mathbf{K}$ is the intrinsic matrix of the camera. (f_x, f_y) and (c_x, c_y) in the intrinsic matrix are the focal length and principal point, which can be known in advance through camera calibration.

Figure 2.1 depicts the perspective projection model in (2.2). To understand the physical meaning, We can divide the transformation into two steps: First, the 3D point $\mathbf{x}_i$ in the object coordinate is transformed to the camera coordinate point

doi:10.6342/NTU202400430

$\mathbf{x}_{ci} = [x_{ci}, y_{ci}, z_{ci}]^\top$ through the extrinsic matrix $\mathbf{T}_{ext}$:

$$\begin{bmatrix} x_{ci} \\ y_{ci} \\ z_{ci} \end{bmatrix} = \mathbf{T}_{ext} \begin{bmatrix} x_i \\ y_i \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} | \mathbf{t} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 0 \\ 1 \end{bmatrix} \tag{2.3}$$

The rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{t}$ are determined by the object's orientation and position, respectively. With both having three degrees of freedom, the pose encompasses a total of 6 degrees of freedom.

Second, the point $\mathbf{x}_{ci}$ in camera coordinate is projected to the point $\mathbf{u}_i$ on the image plane through the camera's intrinsic matrix $\mathbf{K}$. From (2.2) and (2.3), we have

$$u_i = f_x \frac{x_{ci}}{z_{ci}} + c_x, \quad v_i = f_y \frac{y_{ci}}{z_{ci}} + c_y. \tag{2.4}$$

The transformation essentially involves scaling from the actual 3D dimension to the camera image pixel coordinate using (f_x, f_y) and a translation to shift the origin point to the top-left corner of the camera image using (c_x, c_y).

A pose estimation algorithm aims to find a pose $\mathbf{p}$ that minimizes a specific error function based on the observed camera image point $\hat{\mathbf{u}}_i = [\hat{u}_i, \hat{v}_i]^\top$. There are mainly two types of error functions:

- **Reprojection error** measures the image distance between a projected point and an observed one. This form of error is employed in the PnP algorithm.

$$E_r(\mathbf{p}) = \frac{1}{n} \sum_{i=1}^{n} \left((\hat{u}_i - u_i)^2 + (\hat{v}_i - v_i)^2 \right) \tag{2.5}$$

- **Appearance error** computes the appearance difference between the target template and its projection onto the camera frame. This form of error is employed in direct approaches and includes two types: Sum of Absolute Differences (SAD)

$$E_{a_1}(\mathbf{p}) = \frac{1}{n} \sum_{i=1}^{n} |\mathcal{I}_c(\mathbf{u}_i) - \mathcal{I}_t(\mathbf{x}_i)| \tag{2.6}$$

doi:10.6342/NTU202400430

and Sum of Squared Differences (SSD)

$$E_{a_2}(\mathbf{p}) = \frac{1}{n}\sum_{i=1}^{n}\left(\mathcal{I}_c(\mathbf{u}_i) - \mathcal{I}_t(\mathbf{x}_i)\right)^2 \tag{2.7}$$

2.2 Related Work

2.2.1 Marker-Based Pose Estimation

Marker-based pose estimation relies on fiducial markers, which are well-designed patterns with distinctive and robust visual characteristics such as simple binary patterns. Thanks to these distinguishable patterns, fiducial markers can be easily detected and located, leading to efficient and accurate pose estimation. A typical pipeline to recognize the fiducial markers involves thresholding the camera image, extracting the region of the markers, and decoding the patterns. Subsequently, distinctive features such as four corners of the square border [10, 11] or geometry of dots [13, 14] are used in the pose estimation stage. [48] utilized ArUco [10, 11] markers to build a real-time 6-DoF tracking system of a passive stylus that achieves sub-millimeter accuracy. Despite the high accuracy and speed marker systems can achieve, their applications are limited by the necessity of those pre-defined patterns.

2.2.2 Feature-Based Pose Estimation

Feature-based pose estimation depends on natural features in images. The visual features are extracted from both the planar target and the camera frame, and they are matched to establish a set of correspondence between the 3D point of the target and the 2D projection on the frame. Throughout the years, various types of features have been proposed. One of the most classical approaches is SIFT [20], which leverages the Difference of Gaussians (DoG) to extract multi-scale features and describes the keypoints by major orientations in each scale. SURF [21] speeded up the process by utilizing integral images and the Haar wavelet. To achieve even

doi:10.6342/NTU202400430

more acceleration, several binary descriptors are proposed such as BRISK [22], ORB [23], and FREAK [24]. In addition, some RANSAC [25, 26] algorithms are applied to eliminate outliers, enhancing the robustness of feature matching.

After feature extraction and matching, the object pose can be computed by Perspective-n-Point (PnP) algorithms. Numerous PnP algorithms are developed to optimize the pose using the reprojection error described in (2.5), such as EPnP [27] and OPnP [28]. However, the performance of feature-based pose estimation algorithms relies on the successful extraction and matching of features between the planar target and the camera frame. They are less effective when dealing with textureless targets or blurry frames.

2.2.3 Direct Pose Estimation

Direct pose estimation determines the optimal pose from pre-defined candidates by minimizing the appearance error between the planar target and its projection onto the camera frame, which is described in (2.6) and (2.7). Therefore, the 6-DoF pose estimation problem for planar objects can be simplified to 2D template matching, involving iterative [29, 30, 31, 32, 33, 34, 35] and non-iterative [36, 37, 38, 2] approaches.

Direct Pose Estimation (DPE) [2] proposed by Wu *et al*. achieves state-of-the-art accuracy in direct pose estimation for planar objects. It consists of two stages: *Approximate Pose Estimation (APE)* and *Pose Refinement (PR)*.

APE finds an approximate pose in the predefined pose set. Since the difference between the error of two poses can be bounded in terms of a positive value ε [37], they only need to construct an ε-covering [49] pose set with step sizes derived from the bound instead of searching the entire continuous pose space. As the pose set will be extremely large if we want to achieve high accuracy with a small ε, they developed a coarse-to-fine method based on the branch and bound algorithm to reduce the computational cost. Starting from a coarse ε, a pose set is constructed and the best pose in the set, denoted as $\mathbf{p}_b$, along with its corresponding error

doi:10.6342/NTU202400430

$E_{a_1}(\mathbf{p}_b)$ are obtained. Poses with error within the threshold $E_{a_1}(\mathbf{p}_b) + L$ are reserved for the next step, where L is an empirically pre-defined value. Based on the reserved poses, they expand the pose set with a finer ε' and repeat this process until reaching the desired precision.

PR aims to address the pose ambiguity problem [50] and further refine the pose computed by APE. The pose ambiguity arises from multiple local minima in the error function. To deal with pose ambiguity, they select the two stationary points with the smallest error in (2.5) as the candidate poses. Subsequently, the dense image alignment method, which minimizes the SSD error in (2.7) by the LK-based approach [29], is applied to refine both candidate poses. For each candidate pose, the non-linear least squares problem is solved using the Gauss-Newton iteration method. Finally, the candidate pose with a smaller SSD error is selected as the final pose.

While DPE achieves high accuracy, it is notably slow in its execution. Besides, all the algorithms discussed in Section 2.2.2 and Section 2.2.3 are designed for single planar target only. Matching objects occurring in the camera frame with database targets and calculating the poses efficiently and accurately is the primary challenge in multiple planar object pose estimation. Therefore, we leverage the power of object detectors to address this problem.

2.2.4 Deep Learning Object Detectors

Recently, the field of object detection has been overwhelmingly dominated by deep learning approaches. There are mainly three categories of deep learning object detectors: two-stage, one-stage, and transformer-based.

The two-stage object detection network is first proposed by Girshick *et al*. in R-CNN [39], which divides the detection process into two stages: generating region proposals and classifying the regions using CNN features and linear classifier. Fast R-CNN [40] introduces the RoI (Region of Interest) pooling layer to enable end-to-end training of the feature extractor and classifier, requiring only one pass

doi:10.6342/NTU202400430

of the network per image. Faster R-CNN [41] is the first end-to-end trainable detector, which improves the detection speed by replacing the traditional region proposal methods with the Region Proposal Network (RPN).

The one-stage object detection networks utilize Deep Convolutional Neural Networks (DCNNs) to locate and categorize objects simultaneously. YOLO series [42, 43, 44, 45, 4] are the most representative detectors of the one-stage method. They regress and classify the bounding box directly without using region proposals. Notably, one-stage detection networks stand out as the fastest among the three categories.

Transformers have been proven successful in many computer vision tasks. DETR [46] is the first end-to-end object detection model based on transformers. Deformable DETR [47] uses the deformable attention module, which only attends to a small set of keypoints, to achieve a faster convergence rate and higher throughput.

doi:10.6342/NTU202400430

Chapter 3

Proposed Method

In this chapter, we propose a multiple planar object pose estimation system named DetDPE based on object detection and Direct Pose Estimation (DPE) [2], as shown in Figure 3.2. We will start with the training of the planar object detector in Section 3.1. Then, the details of the proposed DetDPE system are introduced in Section 3.2. Finally, we demonstrate the combination of object detector and feature-based pose estimation method in Section 3.3.

3.1 Planar Object Detector Training

Throughout this thesis, we employ YOLOX [4] as our chosen planar object detector. It is noteworthy that the proposed system is independent of the specific object detector chosen. Users can effortlessly substitute the object detector with any state-of-the-art alternative, thereby harnessing the advancements in this rapidly evolving field. In addition, due to the fixed patterns and shapes exhibited by planar targets, their detection is inherently simpler compared to the detection of general 3D objects. Hence, we can exclusively rely on a modest amount of synthetic data to fine-tune a compact pre-trained model, effectively minimizing the training effort for users.

doi:10.6342/NTU202400430

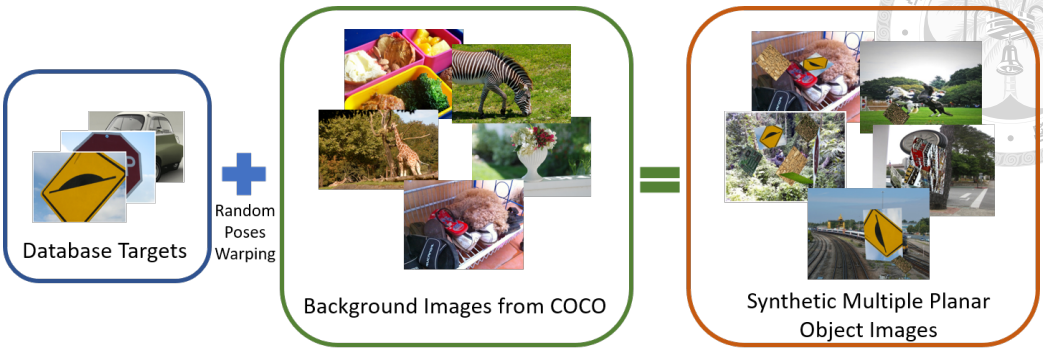


Figure 3.1: The synthetic images for training are generated by warping multiple randomly sampled targets from the database using random poses onto a background image sampled from the MS COCO dataset [1].

3.1.1 Synthetic Data

For fine-tuning the detection model, we only use a small set of synthetic images consisting of 9000 training images and 1000 validation images. Each image is generated by warping multiple randomly sampled targets from the database using random poses onto a background image sampled from the MS COCO dataset [1], as shown in Figure 3.1. Additionally, we introduce random degradation to synthetic images, enhancing the robustness of the detection performance across various conditions: 1) Gaussian blur with kernel size up to 11×11, 2) intensity change with the minimum scale factor set to 0.3, 3) Gaussian noise with the sigma up to 5. We found that these data augmentations are generally sufficient to achieve excellent detection performance in most cases. However, users retain the flexibility to incorporate additional augmentations as needed for different scenarios. Furthermore, we allow partial occlusion between planar objects to address this scenario in real-world situations. This approach also enhances the robustness of pose estimation in the presence of occlusion.

doi:10.6342/NTU202400430

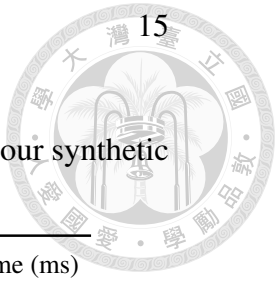


Table 3.1: Performance of different model sizes of YOLOX [4] on our synthetic data

Model	Inference Size	mAP (val)	mAP (test)	Params (M)	Time (ms)
YOLOX-s	480 × 640	96.5	96.2	9.0	6.5
YOLOX-m	480 × 640	97.3	97.1	25.3	9.5
YOLOX-l	480 × 640	97.6	97.4	54.2	15.5

3.1.2 Training and Model Selection

YOLOX offers several versions with different model sizes. Table 3.1 provides a performance comparison of three standard variants on our synthetic data, ranging from YOLOX-s for mobile deployments to YOLOX-l for cloud or high-performance GPU deployments. Each model is initialized with COCO pre-trained weights and fine-tuned for 30 epochs. Given that detecting planar objects is inherently simpler than detecting general 3D objects, all three models can achieve very high mean average precision (mAP). Therefore, in this thesis, we opt for the smallest model, YOLOX-s, to minimize training efforts and enhance the speed of the system. Only 3 GPU hours are needed to fine-tune the model on an NVIDIA RTX 2080 Ti GPU.

3.2 Direct Pose Estimation with Object Detection

The DetDPE system is based on the DPE and integrates with an object detector, as shown in Figure 3.2. We preserve all major steps in DPE while using the detection result to significantly reduce the computational cost. The most time-consuming part of the DPE system is the Approximate Pose Estimation (APE) stage. It has to compute the appearance error of a very large candidate pose set. To address this problem, we feed the bounding box information into the create ε-covering set step to constrain the pose set. We approximate the planar target's center with the center of the bounding box. From (2.3), we know that the camera point of the object's

doi:10.6342/NTU202400430

center point is only affected by the translation vector:

$$\begin{bmatrix} x_{ci} \\ y_{ci} \\ z_{ci} \end{bmatrix} = \begin{bmatrix} \mathbf{R}|\mathbf{t} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}. \tag{3.1}$$

Combining with (2.4), we have

$$t_x = \frac{t_z}{f_x}(u_i - c_x), \quad t_y = \frac{t_z}{f_y}(v_i - c_y). \tag{3.2}$$

Hence, we can directly assign t_x and t_y according to t_z, u_i, and v_i. This reduces the size of the candidate pose set by two orders of magnitude, decreasing it from millions of poses to tens of thousands. In addition, we utilize the bounds of the bounding box to further reduce the pose set. Poses where the projections of the target's corners are too far away from the bounds will be eliminated as well.

For the remaining steps in APE, there are some empirically determined hyperparameters. In [2], they limit the number of remaining poses in the branch-and-bound process. If the number of remaining poses exceeds 27000, they decrease the threshold L described in Section 2.2.3 until the constraint is satisfied. Besides, they set a multiplication factor for expanding the pose set. Theoretically, each dimension should add or subtract a step or remain unchanged, resulting in a $\{-1, 0, 1\}^{6\ dimensions} = 729$ times larger pose set. However, they only select a subset of them, setting the multiplication factor to 81, to reduce the computational cost. This factor can also be cut down since two of the six dimensions are fixed. Otherwise, the pose set will still grow significantly during the coarse-to-fine iteration. In our system, we set the remaining poses bound to 2000 and the multiplication factor to 27, respectively.

Other parts of the original DPE including PR, are completely preserved. Hence, the DetDPE system still maintains high accuracy by utilizing the refinement steps. Algorithm 1 summarizes the whole DetDPE system.

doi:10.6342/NTU202400430

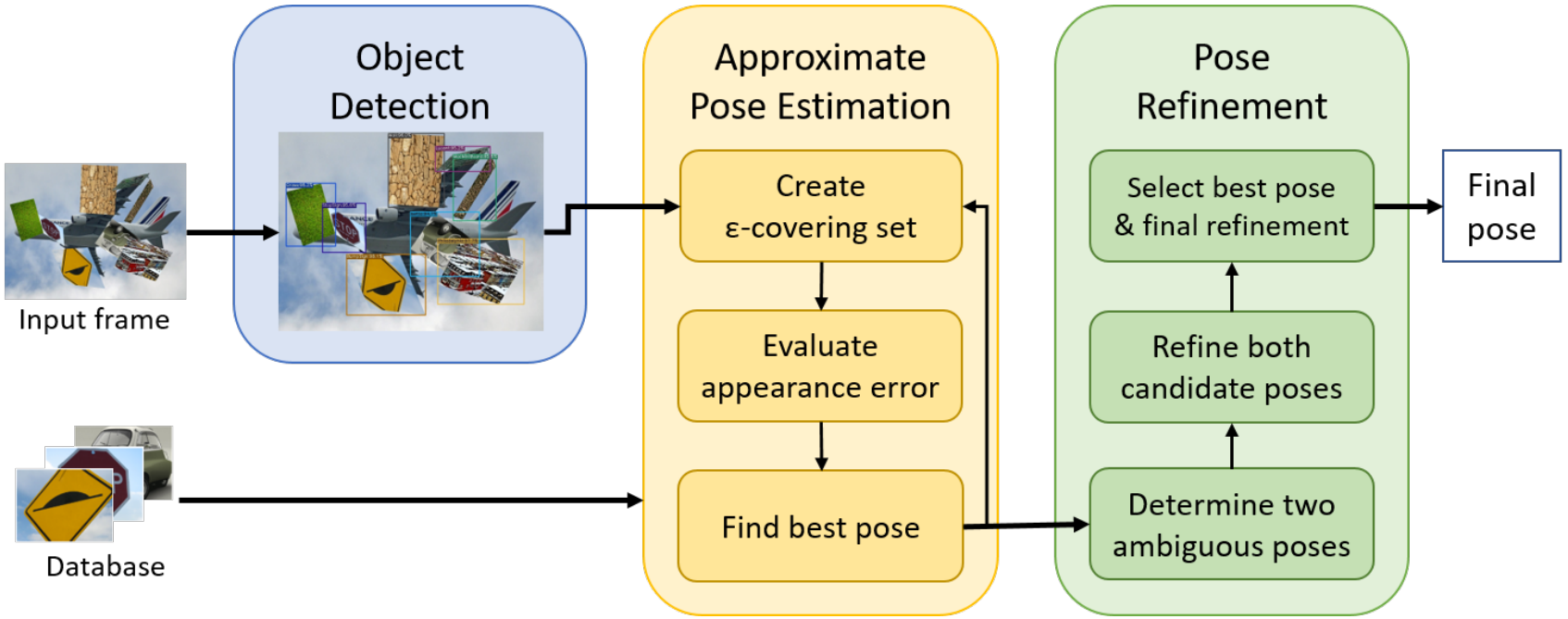
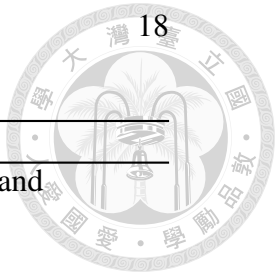


Figure 3.2: **The proposed DetDPE system.** Our system is composed of three stages: object detection, approximate pose estimation (APE), and pose refinement (PR). The bounding boxes obtained from the object detector are used to constrain the candidate pose set in APE to speed up the error evaluation process. The pose from APE is further refined and disambiguated to obtain the final pose.

doi:10.6342/NTU202400430



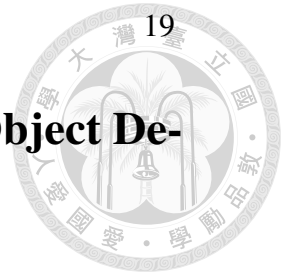
Algorithm 1: Proposed DetDPE Algorithm

Input: Camera image $\mathcal{I}_c$, target image $\mathcal{I}_t$, intrinsic parameters, and parameters ε^*, $\varepsilon_{\Delta\mathbf{p}}$;

Output: Estimated pose $\mathbf{p}^*$;

Object detection for $\mathcal{I}_c$;
for $i = 1 \rightarrow N, N$ is number of object detected **do**
  Build image pyramids for $\mathcal{I}_t$ and $\mathcal{I}_c$ and start from the lowest resolution;
  Create an ε-covering pose set $\mathcal{S}$
  with bounding box constraints imposed;
  Find $\mathbf{p}_b$ from $\mathcal{S}$ according to E_{a_1};
  while $\varepsilon > \varepsilon^*$ **do**
    Obtain the subset $\mathcal{S}_L$ according to $E_{a_1} + L$;
    Diminish ε;
    if Pixel movement of two poses < 1 **then**
      Change to the next image resolution;
    end if
    Replace $\mathcal{S}$ with $\mathcal{S}_L$;
    Find $\mathbf{p}_b$ from $\mathcal{S}$ according to E_{a_1};
  end while
  Determine the candidate poses $\mathbf{p}_1$ and $\mathbf{p}_2$ with $\mathbf{p}_b$;
  for $i = j \rightarrow 2$ **do**
    Let $\mathbf{p}_c = \mathbf{p}_j$;
    repeat
      Compute $\Delta\mathbf{p}$ using Gauss-Newton method;
      $\mathbf{p}_c \leftarrow \mathbf{p}_c + \Delta\mathbf{p}$
    until $\|\Delta\mathbf{p}\| < \varepsilon_{\Delta\mathbf{p}}$
    Let $\mathbf{p}_j = \mathbf{p}_c$;
  end for
  Return the pose $\mathbf{p}^*$ with smaller E_{a_2} from $\mathbf{p}_1$ and $\mathbf{p}_2$;
end for

doi:10.6342/NTU202400430



3.3 Feature-based Pose Estimation with Object Detection

Inspired by the success of DetDPE, we think that the object detection approach can also be integrated with other pose estimation methods, hence being further regarded as a framework for the planar object pose estimation task. Among all types of features mentioned in Section 2.2.2, we chose the most classical one, SIFT [20], to verify our thought.

Figure 3.3 depicts the integration of the feature-based pose estimation method and object detection. First, features are extracted from both the camera image and the database targets. Then we use the bounding boxes generated by the object detector to crop the features of the camera image. Only features located within the bounding boxes are selected, proceeding to the next step. This not only accelerates the feature matching and pose estimation processes but also eliminates a lot of outliers, leading to higher accuracy. Finally, normal feature matching and pose estimation processes are performed to obtain the final pose. Algorithm 2 sums up the approach.

Algorithm 2: Feature-Based Pose Estimation with Object Detection

Input: Camera image $\mathcal{I}_c$, target image $\mathcal{I}_t$, and intrinsic parameters;

Output: Estimated pose $\mathbf{p}^*$;

1: Extract features for $\mathcal{I}_t$ and store in the database;
2: Extract features for $\mathcal{I}_c$;
3: Object Detection for $\mathcal{I}_c$;
4: **for** $i = 1 \rightarrow N$, N is number of object detected **do**
5: Cropping the features of $\mathcal{I}_c$ using the bounding box;
6: Feature matching with cropped features and database features;
7: Compute $\mathbf{p}^*$ by PnP algorithm with RANSAC scheme;
8: Return the pose $\mathbf{p}^*$;
9: **end for**

doi:10.6342/NTU202400430

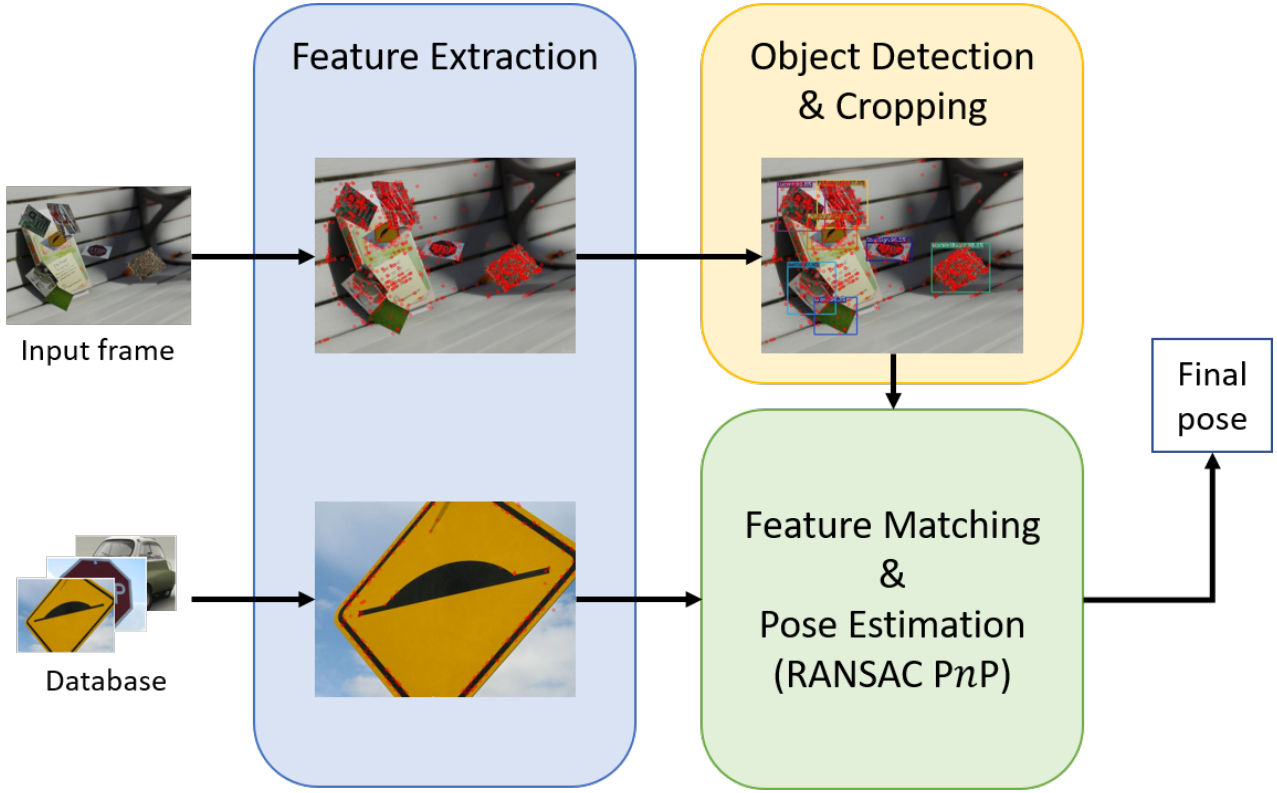


Figure 3.3: The integration of the feature-based pose estimation method and object detection. The extracted features are cropped by the bounding box obtained from the object detector to eliminate most outliers.

doi:10.6342/NTU202400430

Chapter 4

Experiments

We evaluate the proposed DetDPE system on both synthetic and real datasets, including the single target synthetic dataset from [2], Object Pose Tracking (OPT) dataset [3], and our multiple target synthetic dataset. The DetDPE is compared with DPE [2] and the SIFT-based methods. All experiments are conducted on a workstation with an Intel Core i9-9820X 3.3 GHz CPU, 128 GB RAM, and an NVIDIA RTX 2080 Ti GPU.

For error measurement, we use the same metrics as [2]. Given the ground truth rotation matrix $\mathbf{R}_{gt}$ and the translation vector $\mathbf{t}_{gt}$, the rotation error and the translation error are computed by

$$E_{rot}(^\circ) = \arccos\left(\frac{\mathrm{Tr}(\mathbf{R}^\top \cdot \mathbf{R}_{gt}) - 1}{2}\right) \tag{4.1}$$

and

$$E_{tr}(\%) = \frac{\|\mathbf{t} - \mathbf{t}_{gt}\|}{\|\mathbf{t}_{gt}\|} \times 100, \tag{4.2}$$

respectively. A pose is considered successfully estimated if both rotation and translation errors fall within a specific threshold. The error thresholds are defined as $\delta_r = 20^\circ$ for rotation and $\delta_t = 10\%$ for translation. The success rate (SR) is the percentage of the successfully estimated poses. All values of rotation error and translation error presented in this thesis are calculated from successfully estimated poses only.

doi:10.6342/NTU202400430

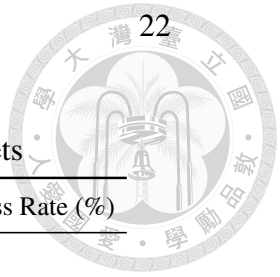


Table 4.1: Performance of YOLOX-s on different datasets

Dataset	Inference Size	Time (ms)	Detection Success Rate (%)
Synthetic (single target) [2]	576 × 768	7.0	99.99
OPT [3]	480 × 800	6.8	99.50
Synthetic (multiple targets)	480 × 640	6.5	98.80

Notably, since previous works do not have a target identification mechanism, they should theoretically match the camera frame with all database targets. This would substantially prolong the experiments. To accelerate the experiments, we provide those methods with ground truth targets for matching. Therefore, all experiments were conducted under the condition of known targets.

4.1 Object Detector Evaluation

We first evaluate the performance of our object detector on different datasets. Table 4.1 shows the evaluation results of our synthetic data fine-tuned YOLOX-s on each dataset. We define the detection success rate here as the percentage of successfully detected targets. Our fine-tuned model can achieve a success rate higher than 98% on each dataset, including the real dataset OPT. This indicates that it is adequate to exclusively employ a small amount of synthetic image data to fine-tune an off-the-shelf compact object detection network for planar object detection in most scenarios. People do not need to expend a significant amount of effort in collecting and annotating training data. Instead, synthetic images and their ground truth can be easily and rapidly generated, minimizing the training effort.

Notably, the results on the multiple target synthetic dataset should theoretically surpass those on the real dataset OPT, given that the testing domain aligns with the training domain. However, there are severe occlusions in some testing images, which cause the degradation of the detection success rate. Nonetheless, the result is still sufficiently accurate.

doi:10.6342/NTU202400430

4.2 Single Target Synthetic Image Dataset

We evaluate our approach on the single target synthetic dataset from [2]. It contains 8400 testing images with 8 planar targets and covers 21 different test conditions, including undistorted images and 20 degraded conditions. The planar targets are categorized into four classes based on their texture characteristics. The size of the images and the planar targets in this dataset are 800×600 and 640×480, respectively. The real dimension of the target is set such that the short side is 1 unit.

4.2.1 Undistorted Images

Table 4.2 shows the evaluation results with the undistorted test images. The proposed DetDPE system performs comparably to the DPE, with both achieving a 100% success rate and outperforming SIFT-based methods. Since we preserve the refinement stage of DPE, DetDPE also demonstrates low rotation and translation error.

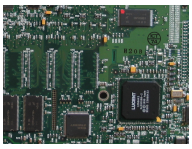
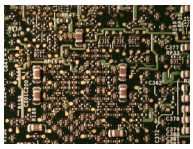
On the other hand, SIFT-based methods are less effective in textureless cases, as few features can be extracted. Nonetheless, with the integration of object detection, the overall performance of the SIFT method improves significantly. This indicates that eliminating outliers by cropping indeed benefits feature-based pose estimation algorithms.

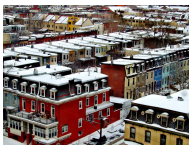
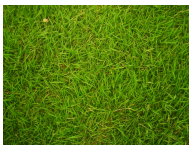
Another phenomenon is that our reduced APE, namely DetAPE, is less precise than the original APE. This is because we reduce the size of the pose set substantially by introducing bounding box constraints and decreasing the number of expanded poses, resulting in our system finding a local optimum instead of a global optimum in the entire pose space. Nevertheless, the approximate poses can still be refined to great accuracy by the PR stage in most conditions.

doi:10.6342/NTU202400430

Table 4.2: Evaluation results with undistorted test images of the single target synthetic dataset [2]. All values of rotation error and translation error are calculated from successfully estimated poses only. The best values are highlighted in red and bold, and the second-best values are highlighted in blue and underlined.

Units: E_{rot}−degree, E_{tr}−percent, SR−percent

	Low Texture						Repetitive Texture					
	Bump Sign			Stop Sign			Lucent			MacMini Board		
												
Method	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR
SIFT	5.16	1.07	*24*	1.93	0.42	*64*	0.59	0.23	*84*	0.22	0.09	*74*
SIFT+detect	4.77	1.13	*24*	2.25	0.71	*72*	0.63	0.27	*92*	0.30	0.18	*88*
APE [2]	1.44	0.45	***100***	1.88	0.41	***100***	0.88	0.46	*98*	1.40	0.64	***100***
DPE [2]	**0.32**	**0.16**	***100***	0.33	**0.20**	***100***	**0.09**	**0.10**	***100***	**0.05**	**0.07**	***100***
DetAPE (ours)	2.04	0.51	***100***	2.55	0.64	***100***	1.47	0.48	***100***	2.06	0.85	***100***
DetDPE (ours)	**0.32**	**0.16**	***100***	**0.29**	0.22	***100***	**0.09**	**0.10**	***100***	**0.05**	**0.07**	***100***

	Normal Texture						High Texture					
	Isetta			Philadelphia			Grass			Wall		
												
Method	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR
SIFT	0.86	0.32	*68*	0.75	0.13	*80*	0.38	0.11	*78*	0.36	0.11	*86*
SIFT+detect	0.86	0.34	*82*	0.78	0.20	*90*	0.41	0.12	*82*	0.49	0.19	*94*
APE [2]	0.99	0.39	***100***	1.32	0.48	***100***	1.42	0.67	***100***	1.02	0.37	***100***
DPE [2]	0.16	**0.15**	***100***	0.09	**0.08**	***100***	**0.09**	**0.10**	***100***	**0.10**	**0.10**	***100***
DetAPE (ours)	1.16	0.45	***100***	1.67	0.48	***100***	2.25	1.15	***100***	1.93	0.52	***100***
DetDPE (ours)	**0.14**	**0.15**	***100***	**0.08**	**0.08**	***100***	**0.09**	**0.10**	***100***	**0.10**	**0.10**	***100***

doi:10.6342/NTU202400430

Table 4.3: Overall results on the single target synthetic dataset [2].

Method	$E_{rot}(°)$	$E_{tr}(\%)$	SR(%)
SIFT	1.23	0.30	*61.57*
SIFT+detect	1.24	0.35	*66.33*
APE [2]	1.42	0.67	***98.43***
DPE [2]	**0.33**	**0.27**	*98.0*
DetAPE (ours)	2.23	0.96	*96.44*
DetDPE (ours)	0.35	0.28	*97.01*

4.2.2 Degraded Images

The dataset contains four different types of image degradation for all planar targets: *Gaussian Blur*, *JPEG Compression*, *Intensity Change*, and *Tilt Angle*. Each type consists of five distortion levels, with a total of 20 conditions. Figure 4.1 and Figure 4.2 are the evaluation results. The proposed DetDPE system still performs similarly to DPE with regard to rotation and translation errors, with only a slightly lower success rate. One thing special is we found that both DetDPE and DPE perform slightly worse than DetAPE and APE in success rate with severe Gaussian blur. As mentioned in Section 2.2.3, the pose ambiguity problem occurs when there are multiple local minima in the error function. For blurry or noisy images, the magnitudes of the two stationary points are even closer. As a result, during the refinement stage, there is a higher probability of selecting the wrong pose.

The SIFT-based method is still inferior to the direct methods. With our detection approach, the success rate can be enhanced by up to 17%.

Table 4.3 shows the overall results on the entire single target synthetic dataset.

4.3 Object Pose Tracking Dataset

Object Pose Tracking (OPT) dataset proposed by [3] is a real dataset for 6-DoF object pose tracking. The 2D object set we use in this thesis contains 138 videos with 20988 frames and 6 planar targets. The conditions of these videos include

doi:10.6342/NTU202400430

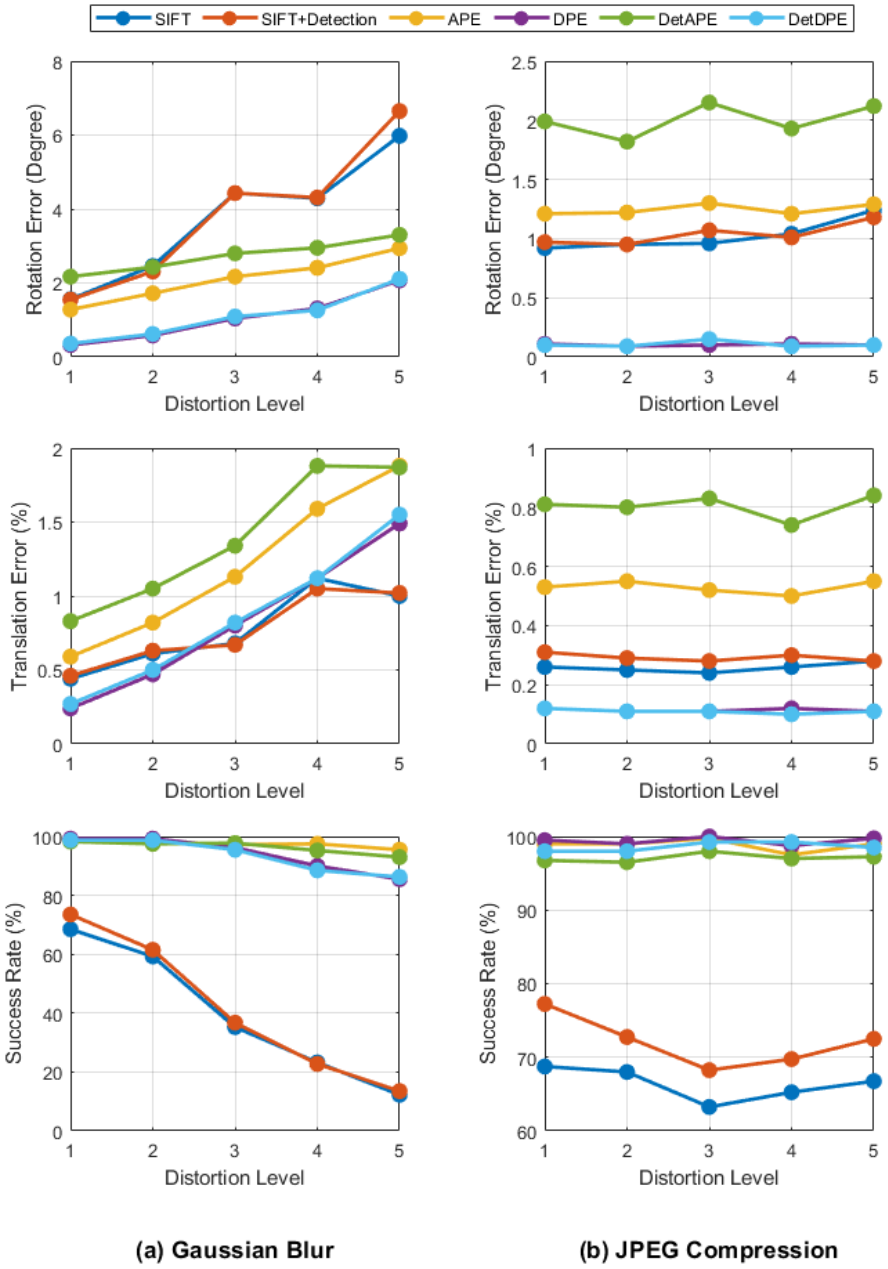


Figure 4.1: Evaluation results on degraded images of the single target synthetic dataset [2] with (a) Gaussian blur and (b) JPEG compression.

doi:10.6342/NTU202400430

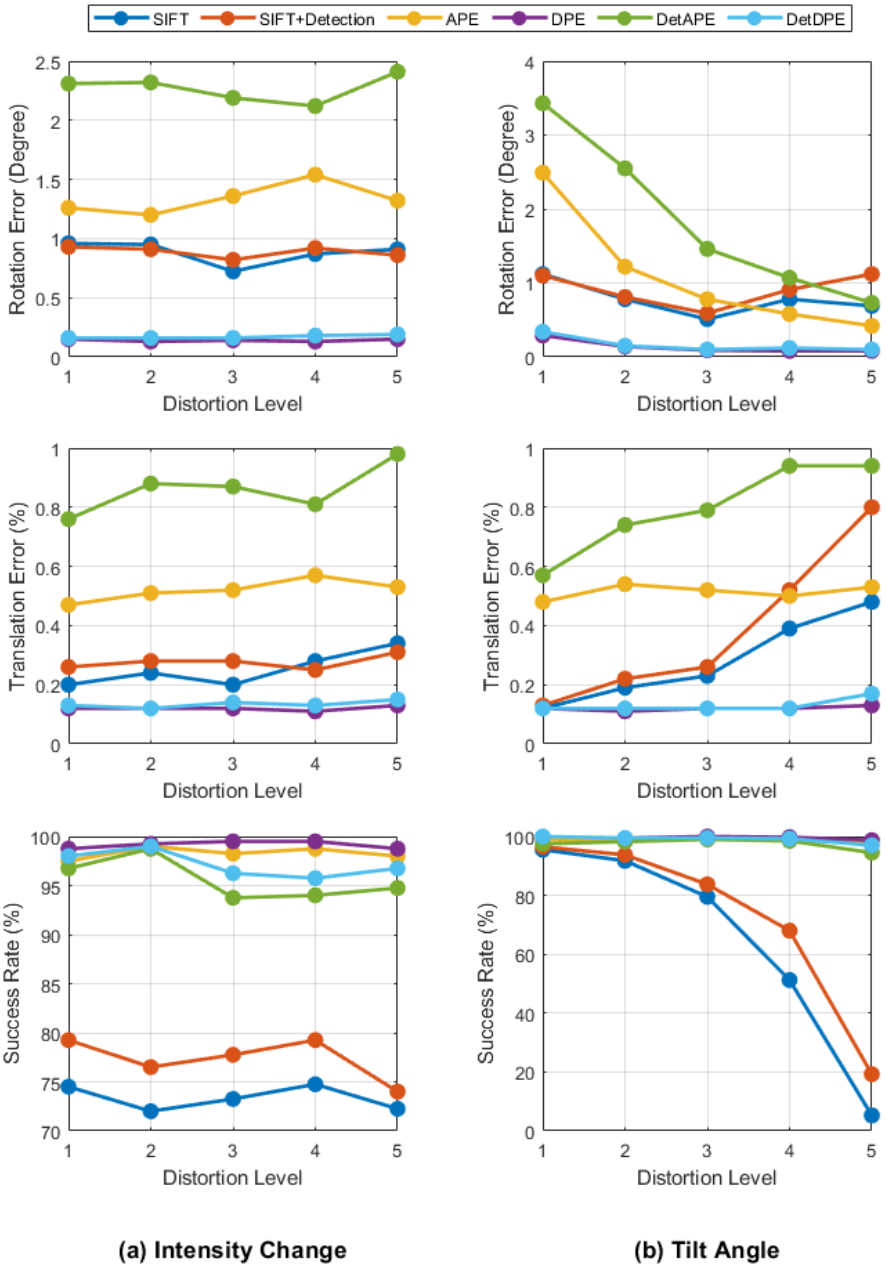
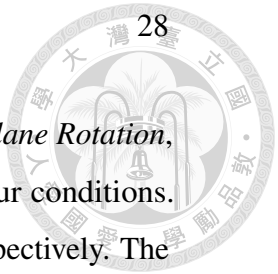


Figure 4.2: Evaluation results on degraded images of the single target synthetic dataset [2] with (a) intensity change and (b) tilt angle.

doi:10.6342/NTU202400430



Flashing Light, *Moving Light*, *Free Motion*, *Translation*, *Zoom*, *In-plane Rotation*, and *Out-of-plane Rotation*. There are 5 speed levels in the last four conditions. The frame size and the target size are 1920×1080 and 300×300, respectively. The real dimension of the planar target is 133.6×133.6 mm^2.

Table 4.4 shows the evaluation results under different conditions. Our DetDPE and DPE exhibit mixed performance, with each having its own strengths and weaknesses. In *Flashing Light* and *Free Motion* conditions, DetDPE is slightly weaker than DPE. However, DetDPE outperforms DPE under *Moving Light* condition. The moving light drastically distorts the color of the image, leading to the failure of the direct method based on appearance error. With object detection, we can recover some performance by constraining the candidate pose set to make it closer to the ground truth.

SIFT-based methods perform well in high-texture target scenes like *Beach*, *Firework*, and *Maple* since they are rich in features. They also outperform direct-based methods as they do not rely on appearance error. We note that all approaches failed with the *Wing* target. The *Wing* target lacks both texture and structural information simultaneously. Therefore, both feature-based and direct-based methods are unable to estimate the pose correctly.

Figure 4.3 and Figure 4.4 show the evaluation results with different speed levels under four motion patterns. Since all approaches fail with *Wing*, their success rates do not exceed 87%. Overall, the proposed DetDPE still performs similarly to DPE and better than SIFT-based methods, except in the *Translation* scenario. Frames in this scenario contain more motion blur. Due to our DetAPE being less precise than APE, DetDPE tends to suffer more against the pose ambiguity problem, resulting in a decreased success rate.

Table 4.5 shows the overall results on the entire OPT dataset.

doi:10.6342/NTU202400430

Table 4.4: Evaluation results on the OPT dataset [3] under different conditions. All values of rotation error and translation error are calculated from successfully estimated poses only. The best values are highlighted in red and bold, and the second-best values are highlighted in blue and underlined.

Units: E_{rot}−degree, E_{tr}−percent, SR−percent

		Wing			Duck			City			Beach			Firework			Maple		
																			
Condition	Method	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR	E_{rot}	E_{tr}	SR
Flashing Light	SIFT	–	–	*0*	6.15	0.56	*57.8*	4.39	0.50	*51.6*	1.71	0.21	***100***	1.76	0.18	***100***	1.66	0.14	***100***
	SIFT+detect	–	–	*0*	5.71	0.50	*62.1*	4.87	0.69	*89.4*	1.63	0.20	***100***	1.76	0.18	***100***	1.66	0.14	***100***
	APE [2]	–	–	*0*	1.57	0.29	***100***	2.93	0.72	***100***	2.12	0.51	***100***	2.04	0.81	***100***	1.93	0.72	***100***
	DPE [2]	–	–	*0*	0.46	0.06	***100***	**0.69**	**0.15**	***100***	**0.46**	**0.10**	***100***	**0.56**	**0.07**	*98.8*	**0.41**	**0.07**	*99.4*
	DetAPE (ours)	–	–	*0*	7.15	1.42	***100***	5.12	1.07	*99.4*	4.78	0.87	***100***	4.53	1.55	***100***	5.42	0.88	***100***
	DetDPE (ours)	–	–	*0*	**0.39**	**0.04**	*94.4*	0.75	0.16	*97.5*	0.47	0.11	*96.6*	0.90	0.14	*91.3*	0.44	0.08	*98.1*
Moving Light	SIFT	–	–	*0*	8.87	0.87	*17.7*	8.27	1.29	*14.0*	1.95	0.24	***100***	**2.31**	**0.28**	***100***	2.45	0.19	***100***
	SIFT+detect	–	–	*0*	9.90	1.01	*21.3*	5.94	0.78	***88.4***	1.95	0.24	***100***	**2.31**	**0.28**	***100***	2.45	0.19	***100***
	APE [2]	–	–	*0*	3.73	0.41	*92.7*	5.37	0.84	*50.6*	4.94	0.50	*71.3*	6.41	1.64	*29.3*	4.03	0.86	*86.6*
	DPE [2]	–	–	*0*	**2.05**	**0.11**	*97.0*	2.25	0.29	*48.2*	**1.60**	**0.09**	*75.0*	3.04	**0.28**	*34.8*	0.78	0.06	*86.0*
	DetAPE (ours)	–	–	*0*	8.71	1.46	*92.1*	7.93	1.27	*47.0*	6.65	0.92	*72.6*	8.73	1.55	*45.1*	6.27	1.09	*92.1*
	DetDPE (ours)	–	–	*0*	2.36	0.12	***99.4***	**2.02**	**0.28**	*48.2*	1.71	0.17	*76.8*	3.44	0.30	*52.4*	**0.75**	**0.05**	*92.1*
Free Motion	SIFT	–	–	*0*	4.83	0.50	*74.2*	5.46	0.95	*59.3*	1.11	0.51	*74.9*	1.22	0.47	*98.0*	2.00	0.42	*99.7*
	SIFT+detect	–	–	*0*	4.66	0.49	*81.0*	3.79	0.65	*86.9*	1.09	0.58	*93.4*	1.20	0.45	*98.1*	2.00	0.42	*99.7*
	APE [2]	**5.20**	**3.14**	***9.57***	2.26	0.40	***100***	1.53	0.91	***100***	1.87	1.43	*83.0*	3.20	2.08	*96.8*	1.80	0.91	***100***
	DPE [2]	7.93	8.14	*1.40*	1.00	**0.18**	***100***	**0.60**	**0.25**	***100***	**0.53**	**0.24**	***100***	**0.57**	**0.31**	***100***	**1.04**	**0.30**	***100***
	DetAPE (ours)	9.90	6.24	*8.80*	9.45	2.81	*87.1*	8.70	3.21	*81.8*	8.07	4.62	*56.2*	9.22	4.11	*36.7*	8.11	3.16	*79.2*
	DetDPE (ours)	7.73	8.42	*1.02*	**0.96**	**0.18**	*98.6*	**0.60**	**0.25**	*95.3*	**0.53**	0.25	*90.2*	0.58	0.34	*94.4*	1.05	**0.30**	*95.0*

doi:10.6342/NTU202400430

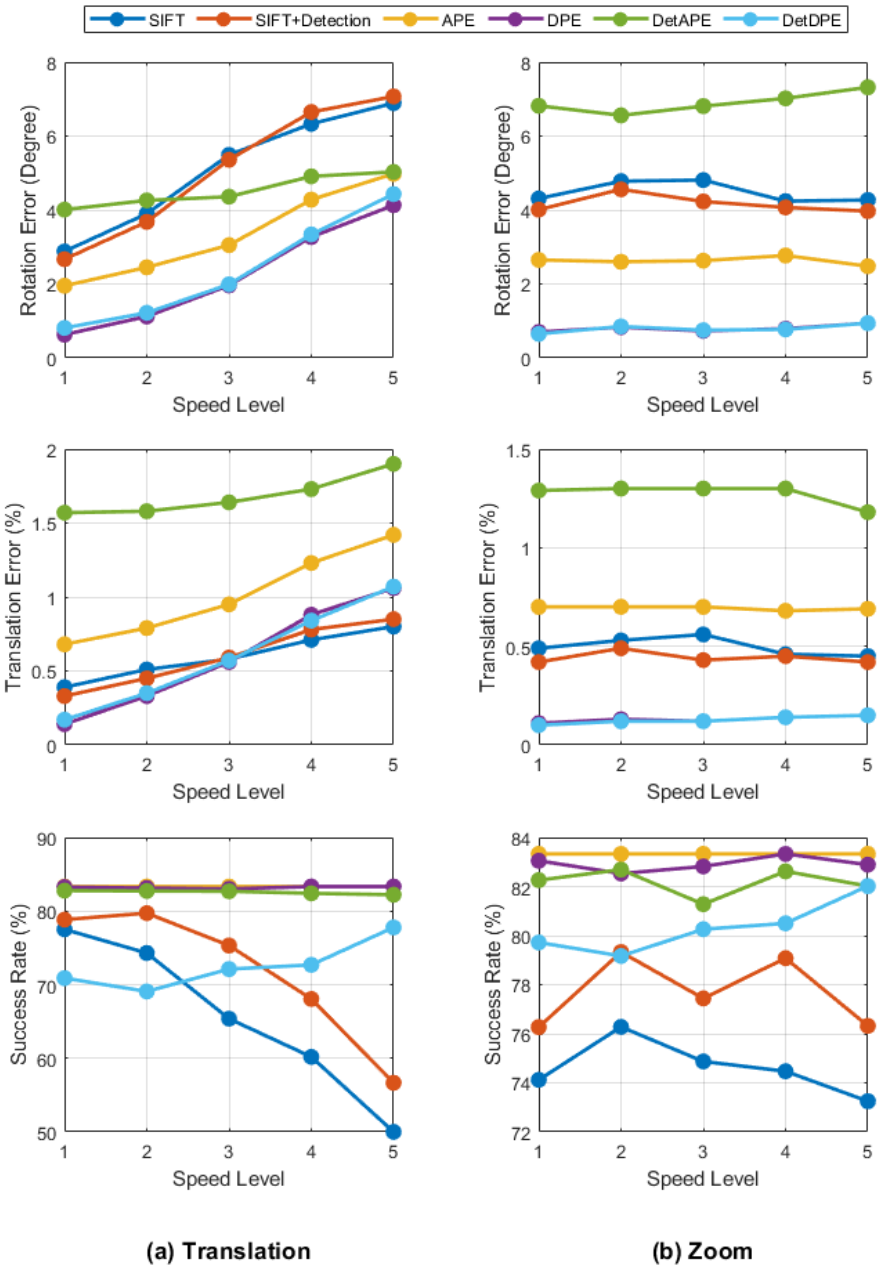


Figure 4.3: Evaluation results on the OPT dataset [3] with (a) translation and (b) zoom.

doi:10.6342/NTU202400430

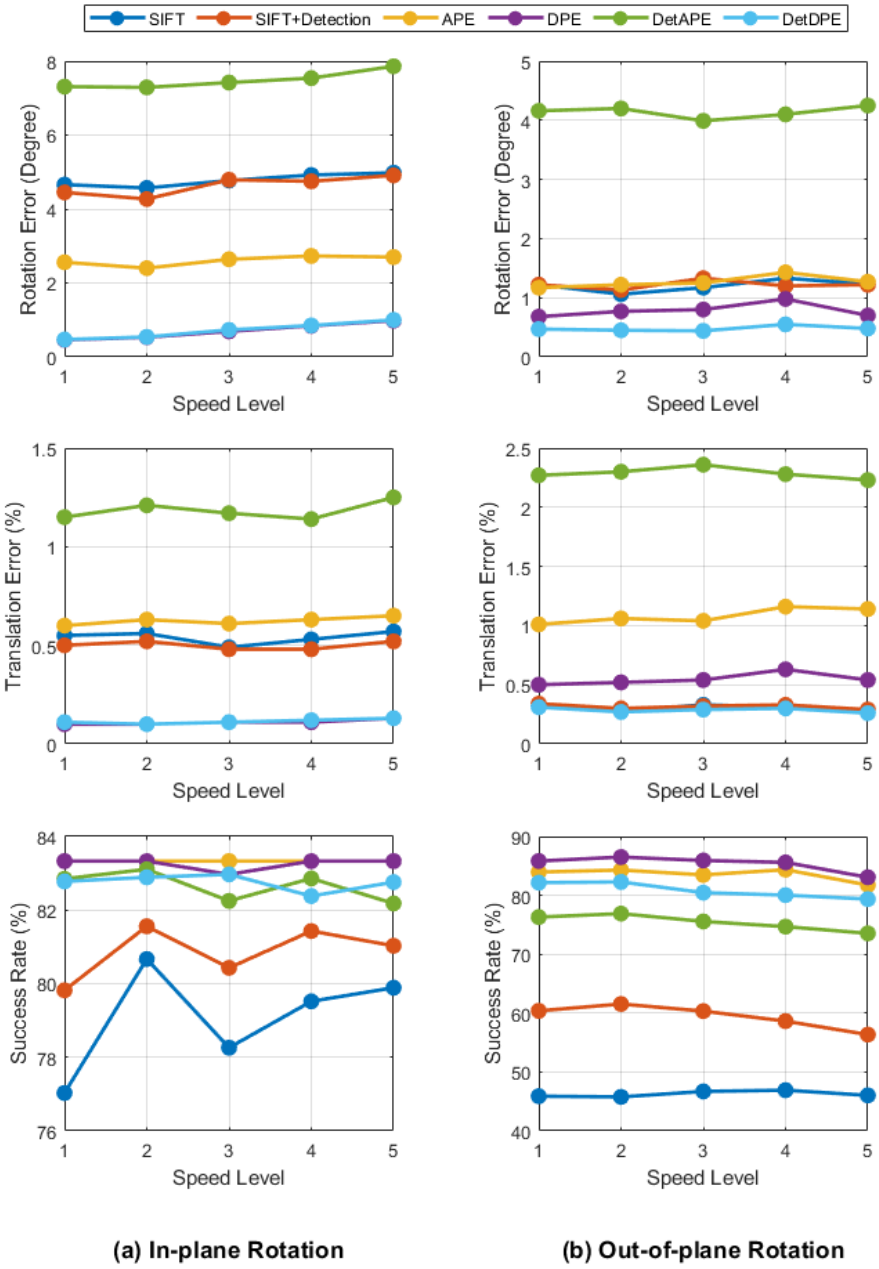


Figure 4.4: Evaluation results on the OPT dataset [3] with (a) in-plane rotation and (b) out-of-plane rotation.

doi:10.6342/NTU202400430

Table 4.5: Overall results on the OPT dataset [3].

Method	$E_{rot}(^\circ)$	$E_{tr}(\%)$	SR(%)
SIFT	3.15	0.46	*63.77*
SIFT+detect	2.93	0.43	*71.88*
APE [2]	2.16	0.91	81.80
DPE [2]	0.81	0.30	***82.14***
DetAPE (ours)	6.02	1.97	*74.35*
DetDPE (ours)	**0.75**	**0.23**	*78.46*

Table 4.6: Evaluation results on the multiple planar target synthetic dataset. All values of rotation error and translation error are calculated from successfully estimated poses only. The best values are highlighted in red and bold, and the second-best values are highlighted in blue underlined. Since SIFT-based methods have really poor success rates, we do not highlight them although their translation errors of successfully estimated poses are low.

Method	$E_{rot}(^\circ)$	$E_{tr}(\%)$	SR(%)
SIFT	10.16	0.70	*3.55*
SIFT+detect	9.78	0.89	*4.04*
APE [2]	3.46	1.95	*56.53*
DPE [2]	**0.88**	**0.91**	*60.66*
DetAPE (ours)	4.88	2.51	*58.57*
DetDPE (ours)	0.98	0.99	***65.91***

doi:10.6342/NTU202400430

4.4 Multiple Target Synthetic Image Dataset

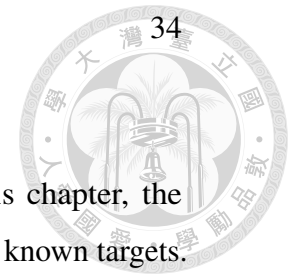
Due to the lack of multiple planar target datasets, we evaluate our system on synthetic images. We generate data using the same planar target database as the single target synthetic dataset [2] and COCO images [1] as background, just as described in Section 3.1.1. We test on 2000 images, containing a total of 8810 targets. Table 4.6 shows the evaluation results. Because the images exhibit various forms of degradation, this dataset is even more challenging. All approaches demonstrate subpar performance compared to the single target dataset, especially SIFT-based methods. Nonetheless, the proposed DetDPE achieves a higher success rate than DPE in this more challenging scenario. While the rotation and translation errors of DetDPE may appear larger than those of DPE, it is essential to note that these errors are calculated from successfully estimated poses only. The results indicate that our DetDPE system is more robust against the multiple target and occlusion scenario since the object detector can identify the locations and regions of the targets, providing more information to the pose estimation stage.

4.5 Runtime Comparison

Table 4.7 shows the average runtime on each dataset. Compared to the DPE, the proposed DetDPE runs 24-27× faster in the APE stage and similar time in the PR stage, combining with a 9-16× faster overall. The runtime of the PR stage is positively correlated to the image size, hence running it slower on the OPT dataset. With the comparable results in previous sections, it can be asserted that our DetDPE system is more efficient than DPE while maintaining high accuracy.

Although the SIFT-based methods are faster than direct-based methods, they have lower accuracy in most conditions. Nonetheless, our detection approach indeed accelerates the feature matching process by 2-4× through cropping and enhances the accuracy as well. Therefore, we can regard this object detection approach as a framework to improve the performance of both direct-based and

doi:10.6342/NTU202400430



feature-based pose estimation algorithms.

It is worth noting that, as mentioned at the beginning of this chapter, the experiments of SIFT and DPE are conducted under the condition of known targets. Thus, the actual execution time will increase proportionally with the size of the database.

doi:10.6342/NTU202400430

Table 4.7: Average runtime (measured in seconds) on each dataset. Values of feature extraction and object detection are computed per image, while others are computed per target.

Method		Dataset (Image Size)		
		Synthetic (single) [2] (800 × 600)	OPT [3] (1920 × 1080)	Synthetic (multiple) (640 × 480)
SIFT	Extraction	0.077	0.247	0.063
	Matching	0.054	0.063	0.050
	PnP	0.001	0.002	0.003
	Total	0.132	0.312	0.116
SIFT+detect	Detection	0.007	0.007	0.007
	Extraction	0.077	0.247	0.063
	Matching	0.027	0.017	0.021
	PnP	0.001	0.001	0.001
	Total	0.112	0.272	0.092
DPE [2]	APE	14.924	12.015	12.603
	PR	0.362	0.872	0.339
	Total	15.286	12.887	12.942
DetDPE (ours)	Detection	0.007	0.007	0.007
	APE	0.539	0.462	0.528
	PR	0.379	0.985	0.290
	Total	0.925	1.454	0.825

doi:10.6342/NTU202400430

Chapter 5

Conclusion

In this thesis, we propose a multiple planar object pose estimation system called DetDPE based on object detection and direct pose estimation. The detector is fine-tuned from an off-the-shelf, pretrained network with only a small amount of synthetic image data, minimizing the training effort when altering the planar targets. In our system, only 3 GPU hours are needed for fine-tuning the selected model, YOLOX-s, on an NVIDIA RTX 2080 Ti GPU. The complexity of the approximate pose estimation stage of the direct pose estimation is reduced substantially by constraining the candidate pose set with detected bounding boxes, resulting in a 24-27$\times$ acceleration. The experimental results demonstrate that DetDPE achieves 9-16$\times$ acceleration overall while maintaining similar accuracy to the state-of-the-art direct-based pose estimation algorithm. In addition, we integrate the detection approach with the SIFT-based pose estimation algorithm, demonstrating up to 8% overall success rate improvement. Consequently, we can regard this object detection approach as a general framework that is capable of enhancing the performance of various types of pose estimation algorithms.

Future work includes:

- Implementing dedicated hardware to achieve real-time performance.
- Designing a general matching algorithm or network that can match all detected targets simultaneously.

doi:10.6342/NTU202400430

- Substituting the fiducial marker system in rigid object pose estimation and localization such as [48] and [51].

doi:10.6342/NTU202400430

Reference

[1] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Proceedings of European Conference on Computer Vision (ECCV)*, 2014, pp. 740–755. iv, 14, 33

[2] P.-C. Wu, H.-Y. Tseng, M.-H. Yang, and S.-Y. Chien, "Direct pose estimation for planar objects," *Computer Vision and Image Understanding (CVIU)*, vol. 172, pp. 50–66, 2018. iv, vi, 3, 4, 10, 13, 16, 21, 22, 23, 24, 25, 26, 27, 29, 32, 33, 35

[3] P.-C. Wu, Y.-Y. Lee, H.-Y. Tseng, H.-I. Ho, M.-H. Yang, and S.-Y. Chien, "A benchmark dataset for 6dof object pose tracking," in *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, 2017, pp. 186–191. iv, v, vi, 21, 22, 25, 29, 30, 31, 32, 35

[4] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," *arXiv preprint arXiv:2107.08430*, 2021. vi, 3, 12, 13, 15

[5] J. J. Rodrigues, J.-S. Kim, M. Furukawa, J. Xavier, P. Aguiar, and T. Kanade, "6d pose estimation of textureless shiny objects using random ferns for bin-picking," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012, pp. 3334–3341. 1

doi:10.6342/NTU202400430

[6] E. Marchand, H. Uchiyama, and F. Spindler, "Pose estimation for augmented reality: A hands-on survey," *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, vol. 22, no. 12, pp. 2633–2651, 2016. 1

[7] Z. Zhou, J. Karlekar, D. Hii, M. Schneider, W. Lu, and S. Wittkopf, "Robust pose estimation for outdoor mixed reality with sensor fusion," in *Universal Access in Human-Computer Interaction. Applications and Services. (UAHCI)*. Springer Berlin Heidelberg, 2009, pp. 281–289. 1

[8] M. Fiala, "Artag, a fiducial marker system using digital techniques," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, 2005, pp. 590–596. 2

[9] E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 3400–3407. 2

[10] S. Garrido-Jurado, R. Muñoz-Salinas, F. Madrid-Cuevas, and R. Medina-Carnicer, "Generation of fiducial marker dictionaries using mixed integer linear programming," *Pattern Recognition*, vol. 51, pp. 481–491, 2016. 2, 9

[11] F. J. Romero-Ramirez, R. Muñoz-Salinas, and R. Medina-Carnicer, "Speeded up detection of squared fiducial markers," *Image and Vision Computing*, vol. 76, pp. 38–47, 2018. 2, 9

[12] B. Benligiray, C. Topal, and C. Akinlar, "Stag: A stable fiducial marker system," *Image and Vision Computing*, vol. 89, pp. 158–169, 2019. 2

[13] H. Uchiyama and H. Saito, "Random dot markers," in *Proceedings of 2011 IEEE Virtual Reality Conference*, 2011, pp. 35–38. 2, 9

[14] H. Uchiyama and E. Marchand, "Deformable random dot markers," in *Proceedings of IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2011, pp. 237–238. 2, 9

doi:10.6342/NTU202400430

[15] L. Calvet, P. Gurdjos, C. Griwodz, and S. Gasparini, "Detection and accurate localization of circular fiducials under highly challenging conditions," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 562–570. 2

[16] D. Hu, D. DeTone, and T. Malisiewicz, "Deep charuco: Dark charuco marker pose estimation," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 8428–8436. 2

[17] J. Peace, E. Psota, Y. Liu, and L. C. Pérez, "E2etag: An end-to-end trainable method for generating and detecting fiducial markers," *arXiv preprint arXiv:2105.14184*, 2021. 2

[18] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes," in *Proceedings of Robotics: Science and Systems (RSS)*, 2018. 2

[19] Y. Hu, P. Fua, W. Wang, and M. Salzmann, "Single-stage 6d object pose estimation," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[20] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision (IJCV)*, vol. 60, no. 2, pp. 91–110, 2004. 2, 9, 19

[21] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding (CVIU)*, vol. 110, no. 3, pp. 346–359, 2008. 2, 9

[22] S. Leutenegger, M. Chli, and R. Y. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 2548–2555. 2, 10

doi:10.6342/NTU202400430

[23] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to sift or surf," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 2564–2571. 2, 10

[24] A. Alahi, R. Ortiz, and P. Vandergheynst, "Freak: Fast retina keypoint," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 510–517. 2, 10

[25] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM (CACM)*, vol. 24, no. 6, pp. 381–395, 1981. 2, 10

[26] O. Chum and J. Matas, "Matching with prosac - progressive sample consensus," in *Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 220–226. 2, 10

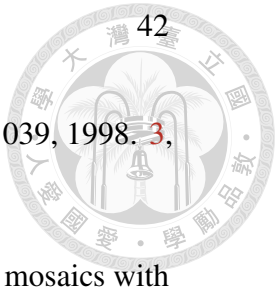
[27] V. Lepetit, F. Moreno-Noguer, and P. Fua, "Epnp: An accurate o(n) solution to the pnp problem," *International Journal of Computer Vision (IJCV)*, vol. 81, no. 2, pp. 155–166, 2009. 2, 10

[28] Y. Zheng, Y. Kuang, S. Sugimoto, K. Astrom, and M. Okutomi, "Revisiting the pnp problem: A fast, general and optimal solution," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2013, pp. 2344–2351. 2, 10

[29] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI)*, vol. 81, 1981, pp. 674–679. 3, 10, 11

[30] G. D. Hager and P. N. Belhumeur, "Efficient region tracking with parametric models of geometry and illumination," *IEEE Transactions on Pattern Analysis*

doi:10.6342/NTU202400430



and Machine Intelligence (TPAMI), vol. 20, no. 10, pp. 1025–1039, 1998. 3, 10

[31] H.-Y. Shum and R. Szeliski, "Construction of panoramic image mosaics with global and local alignment," *Panoramic Vision*, pp. 227–268, 2001. 3, 10

[32] S. Baker and I. Matthews, "Equivalence and efficiency of image alignment algorithms," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, 2001, pp. 1090–1097. 3, 10

[33] E. Malis, "Improving vision-based control using efficient second-order minimization techniques," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, 2004, pp. 1843–1848. 3, 10

[34] A. Crivellaro and V. Lepetit, "Robust 3d tracking with descriptor fields," in *Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3414–3421. 3, 10

[35] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *Proceedings of European Conference on Computer Vision (ECCV)*, 2014, pp. 834–849. 3, 10

[36] Y.-T. Chi, J. Ho, and M.-H. Yang, "A direct method for estimating planar projective transform," in *Proceedings of Asian Conference on Computer Vision (ACCV)*, 2011, pp. 268–281. 3, 10

[37] S. Korman, D. Reichman, G. Tsur, and S. Avidan, "Fast-match: Fast affine template matching," *International Journal of Computer Vision (IJCV)*, vol. 121, no. 1, pp. 111–125, 2017. 3, 10

[38] J. F. Henriques, P. Martins, R. F. Caseiro, and J. Batista, "Fast training of pose detectors in the fourier domain," in *Proceedings of Neural Information Processing Systems (NIPS)*, 2014, pp. 3050–3058. 3, 10

doi:10.6342/NTU202400430

[39] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of 2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587. 3, 11

[40] R. Girshick, "Fast r-cnn," in *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440–1448. 3, 11

[41] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Proceedings of Neural Information Processing Systems (NIPS)*, vol. 28, 2015. 3, 12

[42] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788. 3, 12

[43] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6517–6525. 3, 12

[44] ——, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018. 3, 12

[45] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020. 3, 12

[46] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proceedings of European Conference on Computer Vision (ECCV)*, 2020, pp. 213–229. 3, 12

doi:10.6342/NTU202400430

[47] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2021. 3, 12

[48] P.-C. Wu, R. Wang, K. Kin, C. Twigg, S. Han, M.-H. Yang, and S.-Y. Chien, "Dodecapen: Accurate 6dof tracking of a passive stylus," in *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, 2017, p. 365–374. 9, 37

[49] Wikipedia contributors, "Delone set — Wikipedia, the free encyclopedia," 2017, [Online; accessed 8-May-2018]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Delone_set&oldid=795315991 10

[50] G. Schweighofer and A. Pinz, "Robust Pose Estimation from a Planar Target," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 28, no. 12, pp. 2024–2030, 2006. 11

[51] R. Muñoz-Salinas, M. J. Marín-Jimenez, E. Yeguas-Bolivar, and R. Medina-Carnicer, "Mapping and localization from planar markers," *Pattern Recognition*, vol. 73, pp. 158–171, 2018. 37

doi:10.6342/NTU202400430