UNIVERSITY OF TARTU
Faculty of Science and Technology
Institute of Computer Science
Computer Science Curriculum

Anton Zaliznyi

Real-time Pose Estimation of a Surgical Tool using Optical Coherence Tomography

Master's Thesis (30 ECTS)

Supervisors: Dmytro Fishman, PhD
Lueder Kahrs, PhD

Tartu 2025

Real-time Pose Estimation of a Surgical Tool using Optical Coherence Tomography

Abstract:
Minimally invasive and robotic-assisted surgeries have transformed medicine by reducing patient trauma, infection risk, and recovery times. Within these procedures, precise instrument tracking is critical, especially when navigating intricate anatomical structures and performing delicate interventions such as neurosurgery and microsurgery. Optical coherence tomography has emerged as a promising imaging modality that can provide high-resolution, real-time, and volumetric field-of-view for surgical sites. Existing methods that leverage optical coherence tomography for instrument pose tracking primarily focus on rigid instruments or rely on artificial markers; however, these approaches may fall short in practical scenarios involving occlusions and the dynamic nature of multi-jointed surgical tools. This thesis addressed this challenge by developing a markerless, high-speed, and accurate pose estimation method for an 8-degree-of-freedom microsurgical tool using optical coherence tomography. The proposed method achieves an average position error of 0.26 millimeters, an orientation error of 2.3 degrees, and joint angle errors of 1.9 and 1.9 degrees for θ_1 and θ_2, respectively, while operating with an inference speed of 20 milliseconds per volume. By eliminating the need for markers and being robust to occlusions, our method improves the reliability and feasibility of optical coherence tomography-based microsurgical instrument tracking in complex, dynamic, and realistic surgical environments. Future work should focus on testing this approach with more annotated real-world data and validating its effectiveness through in-vivo applications, thereby enhancing its reliability and practical impact.

Keywords:
Pose estimation, Neural networks, Optical coherence tomography

CERCS: P176 – Artificial intelligence; T111 – Imaging, image processing; B110 - Bioinformatics, medical informatics, biomathematics, biometrics;

Kirurgilise tööriista asendi hindamine reaalajas optilise koherentstomograafia abil

Lühikokkuvõte:
Minimaalselt invasiivsed ja robotite abiga operatsioonid on muutnud meditsiini, vähendades patsiendi traumasid, infektsiooniriski ja taastumisaega. Nende protseduuride puhul on instrumentide täpne jälgimine ülioluline, eriti keerulistes anatoomilistes struktuurides navigeerimisel ja delikaatsete sekkumiste (nt neurokirurgia ja mikrokirurgia) tegemisel. Optiline koherentstomograafia on kujunenud paljulubavaks pildistamisviisiks, mis suudab pakkuda kirurgilistele kohtadele kõrge eraldusvõimega, reaalajas ja mahulist vaatevälja. Olemasolevad meetodid, mis võimendavad optilist koherentstomograafiat instrumentaalpooside jälgimiseks, keskenduvad peamiselt jäikadele instrumentidele või tuginevad tehismarkeritele; need lähenemisviisid võivad siiski osutuda puudulikuks praktilistes stsenaariumides, mis hõlmavad oklusioone ja mitme liigendiga kirurgiliste tööriistade dünaamilist olemust. See lõputöö käsitles seda väljakutset, töötades välja markeriteta, kiire ja täpse poosi hindamise meetodi 8 vabadusastmega mikrokirurgilisele tööriistale, kasutades optilist koherentstomograafiat. Kavandatav meetod saavutab 20 millisekundilise järeldamiskiirusega töötades keskmise asukohavea 0,26 millimeetrit, orientatsiooniviga 2,3 kraadi ning ühendusnurga vead vastavalt 1,9 ja 1,9 kraadi θ_1 ja θ_2 korral mahu kohta. Kaotades vajaduse markerite järele ja olles vastupidav oklusioonidele, parandab meie meetod optilise koherentsustomograafial põhineva mikrokirurgilise instrumendi jälgimise usaldusväärsust ja teostatavust keerukates, dünaamilistes ja realistlikes kirurgilistes keskkondades. Edaspidine töö peaks keskenduma selle lähenemisviisi katsetamisele rohkem annoteeritud reaalmaailma andmetega ja selle tõhususe kinnitamisele in vivo rakenduste kaudu, suurendades seeläbi selle usaldusväärsust ja praktilist mõju.

Võtmesõnad:
Poosi hindamine, Närvivõrgud, Optiline koherentstomograafia

CERCS: P176 – Tehisintellekt; T111 – Pilditehnika; B110 – Bioinformaatika, meditsiiniinformaatika, biomatemaatika, biomeetrika;

Contents

1 Introduction

Minimally invasive surgery and robotic-assisted surgery have transformed medicine, offering important advantages over traditional open surgery. By limiting the exposure of internal tissues, these methods reduce the risk of infection and post-operative complications. Patients experience less pain, require fewer medications, and recover faster [23, 24, 36], translating to shorter hospital stays and improved quality of life.

A fundamental challenge in minimally invasive and robotic-assisted surgeries lies in accurately tracking the pose of surgical instruments. Constantly knowing the precise pose of an instrument in real-time, often with sub-millimeter precision, is crucial to navigate intricate anatomical areas and ensure the accuracy, safety, and efficacy of the procedure [10, 42, 47]. This is particularly important for delicate procedures such as neurosurgery and microsurgery, where precision and control are critical.

Conventional imaging modalities like magnetic resonance imaging and ultrasound either have limitations in tracking surgical instruments with the required precision or can interfere with magnetically actuated microsurgical tools. Optical coherence tomography (OCT) has emerged as a promising alternative, offering micrometer-scale resolution, real-time imaging, and volumetric field-of-view. Already used in ophthalmic [14, 51] and neurosurgical [58, 18] applications, OCT's capability to provide detailed subsurface information makes it particularly suitable for guiding microscale interventions.

Existing methods primarily focus on rigid instruments or rely on artificial markers. However, these approaches may fall short in practical scenarios involving occlusions and the dynamic nature of multi-jointed surgical tools. This thesis focuses on developing a pose estimation solution for a microsurgical tool designed for neurosurgical applications. It addresses the above challenges by presenting a markerless, high-speed, and accurate pose estimation method for an 8-degree-of-freedom multi-jointed microsurgical tool. The proposed method achieves an average position error of 0.26 millimeters, an orientation error of 2.3 degrees, and joint angle errors of 1.9 and 1.9 degrees for θ_1 and θ_2, respectively, while operating with an inference speed of 20 milliseconds per volume. Our approach achieves high accuracy and real-time performance while remaining resilient to occlusions, noise, and tool being partially out of view.

The thesis is organized as follows. The Background section introduces the field of minimally invasive surgery, the fundamentals of OCT imaging, the specific problem addressed, and the hardware setup utilized. It also reviews existing studies, highlighting their contributions and limitations. The Methods section provides a detailed explanation of the deep learning architectures explored, the data acquisition and annotation processes, and the changes implemented to enhance performance. The Experiments and Results section evaluates the proposed methods through comprehensive testing on synthetic and real OCT volumes, demonstrating their effectiveness, robustness, and areas for potential improvement.

2 Background

Surgeries have played a transformative role in medicine, saving countless lives by addressing conditions that are otherwise untreatable. However, surgical interventions come with inherent risks. Post-operational infections and inflammations increase mortality risk [43] even after hospital discharge [13] and require robust infection control protocols [21]. Post-surgery trauma can negatively impact recovery [50], lead to life-long morbidities [9], or even life-threatening consequences [11]. Even when surgeries are technically successful, patients may still suffer from post-operative issues that can lead to extended recovery times or even mortality [2].

Traditional open surgeries, which expose large internal areas, increase the risk of post-operative infections and complications. The invasiveness of a procedure is directly related to the level of risk involved. These risks have driven innovation toward minimally invasive surgeries, which involve smaller incisions, often through the use of advanced imaging and robotic systems. Minimally invasive surgery limits the exposure of internal tissues, lowering infection risks. Moreover, the reduced trauma to the body means patients experience less post-operative pain, require fewer medications, and generally recover faster [23, 24, 36]. Faster recovery times translate to reduced hospital stays and a quicker return to daily activities, improving the quality of life for patients.

One considerable challenge of minimally invasive surgeries is that surgeons require specialized training to master techniques, which often involve delicate hand-eye coordination, depth perception, and precision. Training for complex procedures can take years [30], and even experienced surgeons face a steep learning curve with new technologies [27]. Another important factor is that surgeons must rely primarily on visual cues and indirect imaging, which can make it harder to identify certain issues, such as tissue abnormalities or internal bleeding, which would be more detectable by touch in open surgery [16].

Technological advancements in robotic-assisted surgery have enhanced the precision with which minimally invasive procedures can be conducted [1]. Robotics can aid surgeons in performing complex maneuvers with greater accuracy, reducing the margin of error and improving patient outcomes. This aspect of minimally invasive surgeries is especially beneficial in delicate and intricate surgeries, such as neurosurgery, cardiovascular, or ophthalmic [3, 29, 44, 61]. In addition, robotic systems can filter out tremors and also address the lack of haptic feedback (sense of touch) in minimally invasive surgeries by integrating sensory feedback [16]. Moreover, advanced robotic-assisted surgery systems incorporate AI to analyze intraoperative data, anticipate potential complications, and assist with decision-making [45].

Robotics and advanced imaging technologies facilitate the possibility of telesurgery, where surgeons operate on patients from remote locations. For example, the first telesurgery was performed in 2001, which was conducted on a patient in Strasbourg, France, by a surgical team in New York, USA, more than 6000 km away, using the ZEUS

robotic system (Intuitive Surgical, Sunnyvale, CA, USA). This capability is crucial for providing specialized care to patients in remote or underserved areas, potentially reducing disparities in healthcare access. Robots could also perform life-saving procedures under remote supervision in extreme or inaccessible environments, such as military or space settings [8].

Some surgical procedures, like neurosurgery and microsurgery, demand millimeter or even sub-millimeter precision — levels beyond human capability. In these critical contexts, even slight errors or inconsistencies due to factors such as fatigue, stress, or other human limitations can impact patient outcomes. For example, Neuralink's surgical robot, known as the R1 Robot, is designed to insert ultra-thin, flexible threads into specific regions of the brain - a task that requires exceptional accuracy and control.

Autonomous robotic surgery is a ground-breaking field, which, unlike robotic-assisted surgery where the surgeon maintains control, introduces varying levels of independence. Autonomous systems can perform specific tasks, make decisions, and even complete entire procedures with minimal or no human intervention. Although most autonomous robotic systems are still in the experimental stage, few have been successfully implemented in clinical settings. Procedures such as venipuncture, hair transplantation, intestinal anastomosis, total knee replacement, cochlear implants, and radiosurgery demonstrate impressive capabilities of autonomous surgical systems [45].

A fundamental challenge in minimally invasive, robotic-assisted, and especially autonomous surgery lies in accurately tracking the pose of surgical instruments. Constantly knowing the precise pose of an instrument in real-time, often with sub-millimeter precision, is critical to navigate intricate anatomical areas and ensure the accuracy, safety, and efficacy of the procedure [10, 42, 47]. Surgical tools can have multiple degrees of freedom (i.e., joints), further complicating their control and tracking, especially when they may also be occluded from direct view.

In this study, we address the challenge of pose estimation specifically for neurosurgical applications. We begin by introducing the fundamentals of optical coherence tomography imaging technology, which plays a central role in our research. Next, we detail the surgical robotics setup that serves as the foundation for our approach. Following this, we examine traditional computer vision techniques for solving the pose estimation problem and discuss the advantages of using deep learning instead. We then provide an overview of deep learning principles relevant to our objectives. Finally, we review existing studies in this domain, identifying their limitations and gaps.

2.1 Optical Coherence Tomography

The problem of instrument pose tracking in minimally invasive and robotic-assisted surgeries has been addressed using a range of technologies. Tracking accuracy is often quantified using Target Registration Error (TRE), which represents the positioning error at any point of a tool that is not a marker, most commonly the tool tip [15]. Commercial

optical and electromagnetic tracking systems generally achieve accuracies between 0.2 mm and 1 mm TRE [31]. Optical tracking systems have demonstrated impressive precision, achieving TRE values as low as 0.22 mm in clinical settings [15]. In contrast, electromagnetic tracking, while not requiring a direct line of sight, typically results in a TRE around 0.99 mm [31], which may be insufficient for the precision demanded by delicate procedures such as neurosurgery, ophthalmic surgery and cochleostomy.

In the field of computer vision, researchers have also explored using laparoscopic cameras to estimate surgical tool poses. Although promising, this approach can be heavily affected by occlusions, as well as interference from blood or smoke, leading to variable performance during surgeries [57]. Additionally, other imaging methods, including Magnetic Resonance Imaging (MRI) and Ultrasound, have been investigated for pose estimation [48, 54, 37, 56]. However, these imaging modalities present limitations for minimally invasive surgery. Ultrasound, for instance, lacks sufficient resolution for micrometer-level tracking, while MRI's magnetic interference can hinder procedures involving magnetically actuated tools.

Optical coherence tomography (OCT) has emerged as a promising solution for such high-precision tracking needs. With micrometer-range resolution, OCT provides a detailed, volumetric field-of-view and captures subsurface information based on the reflective properties of different materials [28]. OCT operates by using low-coherence light waves, typically in the near-infrared spectrum, to measure the time delay and intensity of light reflected from internal structures within biological tissues. By directing light into the tissue and analyzing the reflected signal, OCT generates cross-sectional images with micrometer-scale resolution. The data is captured as a series of depth scans (A-scans), which are then combined to form a 2D or 3D volumetric image. This ability to create detailed volumetric images in real-time allows OCT to be particularly useful in surgical contexts, where precise, immediate feedback is crucial for tracking tool positions and interacting with subsurface tissue layers.

OCT's high-resolution capabilities make it suitable for guiding microscale interventions, and it has already been incorporated into operating microscopes for ophthalmic surgeries [14, 51] and neurosurgery [58, 18]. The use of OCT as a tracking system has been further explored in cochleostomy procedures with artificial markers, achieving tracking accuracy within the micrometer range [60]. These advancements highlight OCT's potential as an accurate and reliable imaging modality for instrument tracking.

Despite these promising developments, using OCT as a tracking solution presents challenges. Processing OCT volumes is challenging due to the presence of speckle noise, reflection artifacts, and the inherent complexities of 3D imaging data. Additionally, the application of OCT in procedures outside ophthalmology remains limited, with most research focused on tracking rigid instruments. Gessert et al. [22] demonstrated tool tracking with OCT using pyramidal markers, but their approach required a complex setup to acquire sufficient ground-truth data. Li et al. [34] also utilized OCT to navigate a

magnetically actuated microrobot embedded within a tissue, showcasing OCT's potential for guiding tools in real-time during in-vivo (inside the living body) procedures. However, the challenge remains to expand OCT tracking beyond rigid instruments and into more dynamic applications.

These findings collectively indicate that OCT offers distinct advantages in pose estimation, especially for intricate procedures requiring high accuracy, but there remains a need to explore its applications further and overcome the current limitations.

2.2 Envisioned Robotics Setup for Neurosurgery

We shall now explore the details of the hardware robotics setup, including the microsurgical tool and its intended use case. It is important to emphasize that this work is part of a collaborative project with the author's colleagues from the University of Toronto. The microsurgical tool and the hardware setup described below were developed by the author's colleagues and are not contributions of this thesis. The primary focus of this thesis is the development of a software solution designed to meet the requirements of this hardware setup. To develop such a solution, it is essential to have a comprehensive understanding of the robotic system and its operational constraints, which are described in this section below.

In this work, we utilize a magnetic surgical tool designed specifically for minimally invasive neurosurgery, as the above-mentioned author's colleagues proposed in their recent research [19]. The magnetic tool is structured as a three-link serial robotic mechanism with two adjustable joint angles, θ_1 and θ_2, as shown in Figure 1. θ_1 controls the bending of the tool, ranging from -90 to 90 degrees, while θ_2 controls the opening angle, ranging from 0 to 90 degrees. Link0 remains static, while Link1 and Link2 represent the tool's actuated segments. Measuring only 23 mm in length and 4 mm in diameter, the tool is introduced into the patient via a thin delivery platform, allowing for the use of small incisions. The compact size of the instrument is made possible by a wireless magnetic actuation system, which remotely controls each link's movement. This allows the tool to achieve the necessary range of motion to navigate complex anatomical structures and reach sufficient cutting and grasping forces for neurosurgical procedures despite its miniature dimensions.

To achieve real-time pose estimation for this magnetic tool, we incorporate optical coherence tomography imaging. OCT is compatible with magnetic actuation and can be used during in vivo neurosurgery, capturing high-resolution 3D scans via a distal scanner at the endoscope's tip. This scanner is connected to the surgical site through a fiber optic cable, enabling continuous imaging without interference from magnetic fields. After OCT imaging system has captured a volumetric scan, it is then processed on a computer to estimate the tool's exact pose within the patient's anatomy. The computed pose parameters - the tool's position, orientation, θ_1 (bending), and θ_2 (opening) angles - provide the necessary feedback to enable the fine control required for minimally invasive

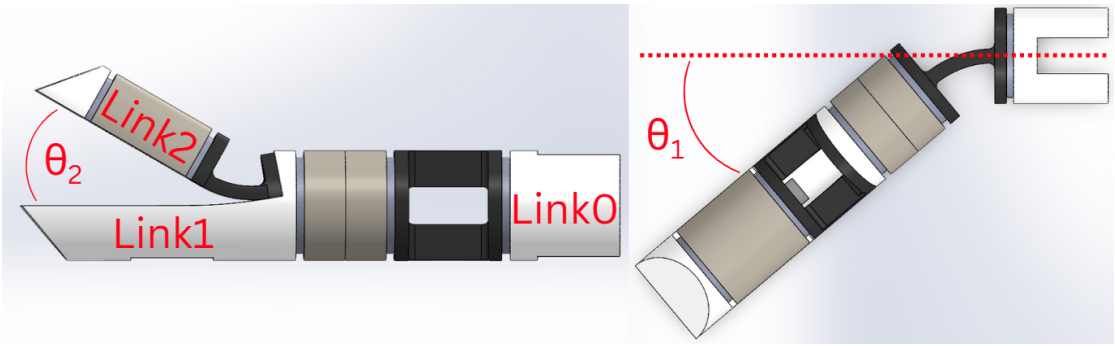

Figure 1. View of the magnetic surgical tool with labeled links and joint angles. The tool, designed for minimally invasive neurosurgery as proposed in [19], consists of three links: Link0 (static base), Link1 (bending segment), and Link2 (opening segment). θ_1 controls the bending of Link1, while θ_2 controls the opening and closing of Link2.

neurosurgical procedures, which demand high precision and safety. The complete envisioned setup is displayed in Figure 2.

The OCT system used in this study is the OMES 4D MHz-OCT System (OptoRes GmbH, Munich, Germany), which operates at a center wavelength of 1310 nm. This high-speed system generates a 3D volumetric scan represented as a numpy array with dimensions of 1096x1936x1152, totaling approximately 2.4 gigabytes. Each volume is acquired in 0.7 seconds, with voxel dimensions of 15.9x13.5x9.4 μm^3. The OCT imaging workspace is 17.4x26.1x10.8 mm^3, although the depth range is limited; beyond 6 mm, the signal drops completely, making deeper structures invisible. For real-time feedback, the OCT scanner settings can be adjusted to output smaller volumes of 28x1936x1152, which are acquired in just 0.05 seconds (equivalent to 20 Hz). This configuration allows for rapid 3D imaging in real-time, with the OCT system capping at 20 Hz, meaning that subsequent processing (e.g., pose estimation) must keep pace at a minimum of 20 Hz. Faster processing speeds are unnecessary, as the overall system operates at this acquisition rate limit.

To summarize the requirements, this study aims to develop a method that, given an OCT volume of size 28x1936x1152, will compute position, orientation, θ_1 (bending), and θ_2 (opening) angles in less than 0.05 seconds (20 Hz). The approach must be robust to noise, other objects in the scene, and occlusions.

Now that we have examined the hardware setup, the microsurgical tool, and the requirements for real-time pose estimation, we shift our focus to the development of a software solution. In the following sections, we will explore existing approaches for pose estimation, their advantages and drawbacks to identify the most suitable methods for addressing the abovementioned challenges.

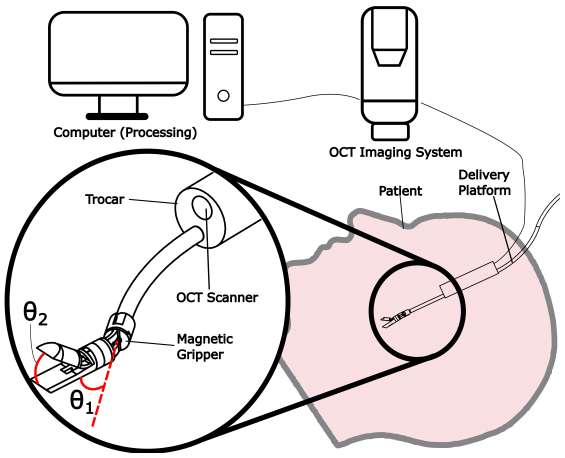

Figure 2. Overview of the surgical setup for in-vivo neurosurgery. The magnetic surgical tool is inserted through a thin delivery platform and operates within the patient's anatomy. The OCT Imaging System provides real-time scans of the surgical site. The scans are processed on the Computer to estimate the tool's pose parameters. The tool is actuated through an external coil system. Image adapted from our previous study [20].

2.3 Pose Estimation with Traditional Computer Vision

In our previous study [20], we developed a marker-based computer vision algorithm to estimate the joint angles of the surgical tool described previously using OCT imaging. The tool had spherical markers strategically placed on its links, with each marker aiding in the detection of specific joint angles.

The process began with pre-processing the raw OCT volumetric data through down-sampling and thresholding to enhance marker visibility. These spherical markers, being 1 mm in diameter, were detected in the OCT volume using a 3D template matching approach. This method involved sliding a 3D template of the marker shape across the OCT volume to identify regions that matched the marker's profile. Since markers consistently appeared as spheres in the OCT scans, the algorithm could reliably detect them without needing rotational adjustments to the template. To reduce multiple detections of the same marker, which could arise from template matching, producing several high-signal matches for each marker, a distance-based clustering algorithm DBSCAN [17] was used. DBSCAN grouped nearby detections, selecting the strongest signal as the true center of each marker. Finally, we apply a post-filtering step to further eliminate false positives among the detected markers. Since we have a computer-aided design (CAD) model specifying the exact placement and spatial relationships between markers, we can use this information to validate the detections. By evaluating whether the detected markers align with the expected configuration and distances defined by the CAD model, we can filter out detections that do not match any plausible marker arrangement, effectively

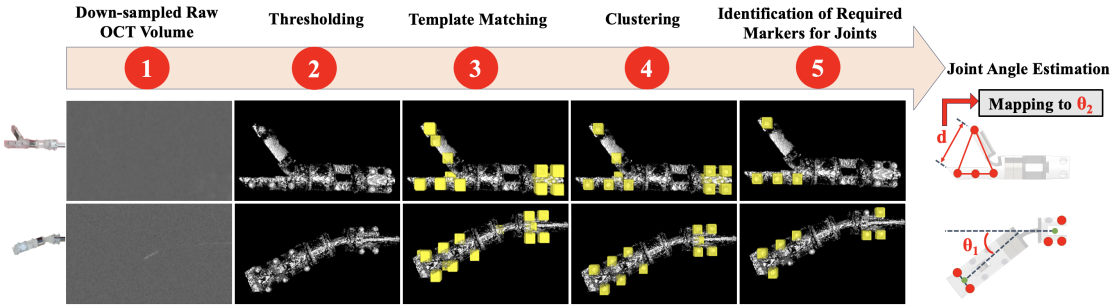

Figure 3. Summary of the approach presented in our previous study [20]. The solution employed traditional computer vision techniques, including thresholding, template matching, and clustering, for marker-based pose estimation of the magnetic surgical tool.

removing remaining false positives. Once the markers were identified, the joint angles were estimated by calculating the spatial relationships between detected markers. For example, the first joint angle θ_1 was determined by projecting detected markers onto a plane and measuring the orientation between them. The second joint angle θ_2 was estimated using distance-based calculations between specific markers on different links. This technique yielded an average error of 2.2° for θ_1 and 2.0° for θ_2, with each estimation process completed within approximately 0.5 seconds on a standard CPU. The complete pipeline of this solution is given in Figure 3.

While effective, this traditional computer vision approach has several limitations. The reliance on markers makes it highly susceptible to occlusions caused by blood, smoke, other objects, self-occlusion, or simply the tool being partially out of view. If even a single required marker is undetected, the entire system fails to estimate the pose accurately. Additionally, this approach requires meticulous marker placement during manufacturing, as even a slight error can lead to inaccuracies, given the system's dependence on precise inter-marker distances. Although this issue can be mitigated by performing calibration and recalculating marker distances after placement, it still demands precision. Furthermore, when switching to a different surgical tool, new markers would need to be carefully positioned according to the same constraints for the algorithm to function effectively. Detecting these markers in noisy OCT volumes also presents a challenge, as the algorithm may falsely identify non-marker objects with similar shapes as markers. Although this method achieves precise angle estimations in controlled conditions, it lacks adaptability for complex or dynamic scenarios where markers may become occluded or where markerless tracking would be advantageous. These limitations highlight the need for deep learning-based pose estimation techniques that could provide more robust, flexible solutions without relying on physical markers.

2.4 Deep Learning Overview

Deep learning has transformed the field of artificial intelligence by enabling machines to learn complex patterns directly from data. At its core, deep learning leverages neural networks with multiple layers to extract and represent features at varying levels of abstraction. Each layer builds upon the previous one, capturing increasingly complex patterns as data flows through the network.

Training deep learning models involves an optimization process known as gradient descent, where the network's parameters are adjusted to minimize a loss function that quantifies the difference between the model's predictions and the actual target values. During training, the gradients of the loss function with respect to each parameter are calculated using backpropagation and then adjusted in the direction that reduces the error. Activation functions, such as ReLU (Rectified Linear Unit) and sigmoid, introduce non-linearity into the network, allowing it to approximate more complex functions beyond linear mappings. This combination of multi-layer architectures, activation functions, loss functions, and gradient-based optimization forms the backbone of deep learning systems.

Deep learning gained traction due to the availability of large datasets and advancements in hardware, particularly GPUs, which allow for parallel processing of matrix operations. Today, deep learning powers a wide array of applications, including image recognition, language processing, and medical image analysis, demonstrating its versatility and effectiveness in capturing complex data patterns.

In this work, deep learning enables robust processing of optical coherence tomography volumes to predict the surgical tool's position, orientation, and joint angles. Unlike traditional methods reliant on markers or handcrafted features, deep learning provides greater resilience to noise, occlusions, and dynamic surgical scenarios, making it a powerful solution for such applications.

2.4.1 Convolutional Neural Networks

One breakthrough in deep learning came with the introduction of Convolutional Neural Networks (CNNs), particularly for image processing tasks. First popularized by LeNet-5 [33] and later advanced by architectures like AlexNet [32], CNNs introduced convolutional layers that could automatically learn spatial hierarchies of features, making them highly effective for visual tasks such as classification, object detection, and segmentation. By using filters that slide across an image to detect patterns, CNNs can recognize different levels of abstraction, from simple edges in early layers to complex structures in deeper layers.

CNNs are composed of several types of layers, each contributing to their success in visual tasks. Convolutional layers apply filters to local patches of the input, capturing spatial relationships through weighted connections. Each filter, or kernel, detects specific features such as edges, textures, or any abstract patterns. This spatial feature extraction

makes CNNs highly effective for identifying relevant image components. Following the convolutional layers, pooling layers reduce the spatial dimensions of the image, retaining the most important features while minimizing computational load and promoting generalization by making the network more robust to minor spatial variations.

Over time, CNN architectures have evolved to incorporate deeper layers and advanced components, leading to increasingly complex and powerful models. VGGNet [35] demonstrated that increasing the network depth could improve performance, while ResNet [25] introduced residual connections to address the vanishing gradient problem, enabling networks with hundreds of layers to be trained effectively. This ability to create "deep" networks without degradation in training has been crucial for tasks requiring fine-grained feature extraction.

2.4.2 Transformers

Transformers [53] have transformed computer vision field by leveraging a self-attention mechanism that captures long-range dependencies, as demonstrated by architectures like Vision Transformer (ViT) [12] and Detection Transformer (DETR) [5].

While transformers are powerful, they come with limitations that make them less practical for our OCT-based pose estimation task. First, transformers require extensive data for effective training due to their large number of parameters and lack of inherent spatial biases. In contrast, our problem involves limited real data, making it challenging to meet transformers' data demands. Second, transformers tend to be computationally expensive and slower in inference, which is problematic in scenarios requiring real-time performance.

Due to their high computational cost and extensive data requirements, transformers are not well-suited for OCT-based 3D pose estimation. CNNs, with their efficient spatial structure, are better suited to handle relatively small data sets and real-time constraints typical of medical applications. Consequently, this work focuses on the use of CNNs to achieve accurate and efficient pose estimation.

2.4.3 Transfer Learning

Transfer learning is a machine learning approach where a model trained on one task is adapted or fine-tuned for a different but related task. It leverages the knowledge gained from a large dataset (source task) to improve performance on a smaller dataset (target task). For example, models pre-trained on extensive image datasets like ImageNet can recognize basic features such as edges, textures, and shapes, reducing the need for training from scratch on new data. This technique is especially valuable in situations like ours, where labeled data is scarce. In our work, we pre-train models on synthetic OCT data and then fine-tune them on real data to bridge the gap between the two domains. Further details will be provided in the following sections.

2.5 Pose Estimation using Deep Learning

We shall now discuss different approaches for pose estimation using deep learning. Although our final goal is to estimate the tool's pose parameters (position, orientation, angles), it is also possible to apply keypoint estimation methods to predict keypoints from which the pose parameters can be calculated. A similar idea of using keypoints as an intermediate representation for pose estimation has been done by Pavlakos et al. [40]. The approach utilizes Hourglass Network [38] explained below to predict keypoints, which are then optimized using a deformable shape model, which allows to estimate the 6-DOF (degree of freedom) pose of an object.

We shall start by exploring the history of human pose estimation as it laid the groundwork for techniques that can be adapted to track other objects, such as surgical tools. DeepPose by Toshev and Szegedy [52] was one of the first deep learning approaches to human pose estimation, pioneering the direct regression of body joint coordinates using convolutional neural networks. This method demonstrated that CNNs could effectively predict human joint positions, marking a departure from traditional methods that relied on hand-crafted features. Stacked Hourglass Networks by Newell et al. [38] further advanced the field with a multi-scale, symmetrical architecture that refines predictions by repeatedly downsampling and upsampling feature maps within an hourglass structure. This approach enabled multi-scale feature extraction and integration, improving pose estimation accuracy by leveraging both local and global spatial context. Building on these advancements, OpenPose by Cao et al. [4] introduced Part Affinity Fields to associate body parts across multiple individuals within an image, facilitating robust multi-person pose estimation.

In RGB-D-based pose estimation, recent advancements have increasingly focused on leveraging point cloud representations to capture fine-grained 3D spatial information, enhancing the accuracy and robustness of pose estimation in cluttered environments. DenseFusion by Wang et al. [55] introduced a dense pixel-wise fusion of RGB and depth features for 6-DOF object pose estimation. By merging color and depth information at a per-pixel level, DenseFusion creates a comprehensive feature representation that allows it to effectively handle occlusions and background clutter, a challenge frequently encountered in real-world scenes. Building on this foundation, PVN3D by He et al. [26] integrates point cloud and voxel-based features to improve 6-DOF pose estimation further. PVN3D employs a point-wise voting scheme for keypoint prediction, followed by feature aggregation across both RGB and depth channels, which enhances robustness in complex environments with occlusion. These approaches highlight the growing preference for point cloud and voxel methods in RGB-D pose estimation, demonstrating that these representations are particularly well-suited for precise 3D spatial understanding in challenging scenarios.

Alternative techniques focus on projecting depth data into a 3D voxel grid. V2V-PoseNet by Moon et al. [6] adopts a voxelized 3D grid as input and uses a fully 3D

convolutional neural network to estimate per-voxel likelihoods for each joint keypoint. This voxel-to-voxel approach, implemented with an Hourglass network architecture, overcomes the challenges of perspective distortion seen in 2D depth images, allowing the network to process 3D spatial structures more accurately and efficiently. Expanding on this voxel-based strategy, Virtual View Selection by Jian Cheng et al. [7] introduces a virtual multi-view approach. This method reprojects a single depth image into multiple virtual viewpoints, the best of which are then selected and fused to enhance 3D hand pose estimation. By focusing on the most informative views, the model addresses occlusions and viewpoint variations more effectively, making multi-view fusion a robust solution for capturing complex 3D structures from limited input.

There are few studies that addressed pose estimation using OCT. Zhou et al. [61] proposed a method using OCT for 6-DOF needle pose estimation in robotic-assisted vitreoretinal surgery, leveraging 3D point cloud segmentation and a modified iterative closest point algorithm. Their approach focused on segmenting needle point clouds from OCT volumes, followed by aligning these with a CAD model to compute pose, achieving high positional accuracy (within 10 μm) in ex-vivo pig eye trials. Overall, the algorithm requires approximately 500 ms to make a prediction, which exceeds our performance requirements. Additionally, Zhou et al. reduce the typical 6-DOF in pose estimation to a 2-DOF optimization problem by constraining the shift and rotation along and around the needle's axis. This simplification assumes that the needle primarily moves along its axis in the surgical context. Although this reduces computational load, it may restrict the method's applicability in scenarios where the tool exhibits more complex or unpredictable movements.

Gessert et al.[22] utilized deep learning for OCT-based pose estimation, proposing Inception3D and ResNeXt3D, 3D CNN architectures specifically tailored to process volumetric OCT data. These models were trained to directly predict 6-DOF pose from OCT volumes of miniature markers. By exploiting volumetric information through a 3D CNN, the approach surpassed typical 2D or 2.5D depth-based methods in accuracy, highlighting the advantages of using OCT's full volumetric data for high-precision pose estimation in microscale environments. Gessert et al. believed that a marker-based system has the advantage that the neural network only needs to be trained once on a specific marker geometry, allowing it to be used across different surgical tools. In contrast, a markerless approach would require retraining for each unique tool. However, as mentioned above, a drawback of marker-based tracking is that if even one marker is undetected (for instance, tools with multiple joints require multiple markers), the entire system fails to estimate the pose. As already explained, in real surgical environments, it is not uncommon for small markers to be occluded by blood, smoke, or the instrument itself. By comparison, a neural network in a markerless system can infer the tool's pose based on visible parts, even if portions of the instrument are temporarily occluded.

Schluter et al. [46] introduced an innovative setup to track markerless 6-DOF surgical

tools. They used a motorized OCT system that dynamically adjusts the field-of-view to follow the instrument's movements, supported by a MOSSE filter for adaptive template matching. This configuration enables the system to estimate both translational and rotational motions by tracking multiple localized points on the instrument. While this mechanical setup allows for precise tracking within the OCT's limited field-of-view, it is constrained by the physical response time of the motorized adjustments, which can hinder real-time performance, particularly for fast or complex rotations. Additionally, the approach is sensitive to occlusions; if critical tracking points become obscured, the system can lose track of the instrument's pose.

Current research focuses on rigid instruments or relies on markers, with few studies addressing real-time pose estimation under realistic conditions where occlusions may occur. The contribution of our work is a **markerless, high-speed, accurate** approach for real-time volumetric pose estimation of **8-degree-of-freedom multi-jointed** microsurgical tool using OCT data that is robust to occlusions, noise, and other objects present in the scene, as well as to the surgical tool being partially out of view. In the next chapter, we discuss the specific methods and technical details used in our proposed solution.

3 Methods

In this section, we detail promising approaches for estimating an object's pose from 3D data. We evaluate two studies, Gessert et al.'s Inception3D and ResNeXt3D, as well as Moon et al.'s V2V-PoseNet, which have demonstrated strong performance on 3D data in related domains. These models serve as baselines that we adapt and modify in future sections to suit the challenges of our particular use case. We also describe the synthetic data generation, data collection, and annotation methods used to power our experiments.

3.1 Gessert et al.'s Inception3D and ResNeXt3D

Gessert et al. [22] is one of the few studies that explore the application of deep learning for pose estimation in OCT volumes. They employed multi-output regression to predict both the position (x, y, z) and orientation (r_x, r_y, r_z) values of the marker. Their findings demonstrate that utilizing 3D volume data yields higher accuracy compared to using multiple 2D projections. Additionally, their models proved robust in the presence of additional objects in the scene.

In their work, they introduced several deep learning model architectures, notably Inception3D and ResNext3D, which extend their 2D counterparts into the 3D domain. Inception3D extends the original Inception design by incorporating 3D convolutions. It uses multi-path convolutional blocks with different kernel sizes to extract features at multiple scales. Additionally, the network integrates residual connections and bottleneck layers for improved training stability and reduced parameter count. ResNeXt3D, inspired by ResNeXt (in 2D), simplifies multi-path designs by using grouped convolutions. Grouped convolutions split the input channels into separate groups, perform convolutions independently within each group, and then concatenate the outputs. This reduces the number of parameters and computations compared to standard convolutions. Also, unlike Inception3D, all paths in ResNeXt3D are identical, which reduces the complexity of architecture tuning.

One key argument from Gessert et al. is that their models, optimized specifically for the marker's geometry, can generalize across different surgical tools without requiring retraining, provided the marker remains the same. However, in our previous research, we identified several issues with relying on markers attached to microsurgical instruments. Markers can become occluded by other objects or move out of the field-of-view, causing the system to fail in pose prediction. We believe these problems are likely to arise frequently when using OCT. Moreover, the physical properties of markers may interfere with the robotic workflow. To address these limitations, we propose a markerless approach in which the full pose of the instrument is regressed. Although this requires retraining for each new surgical tool, it allows the model to learn the tool's geometry, making it more robust to occlusions and partial visibility of the instrument.

We have re-implemented the architectures proposed by Gessert et al. using PyTorch,

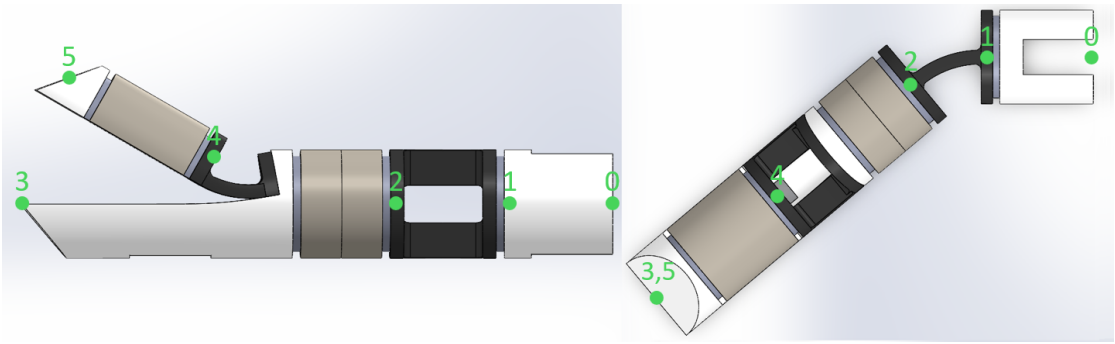

Figure 4. Proposed keypoint placement on the magnetic surgical tool. The figure illustrates six keypoints labeled 0 through 5, which are used for markerless pose estimation. The keypoints allow us to compute the tool's pose parameters.

referencing the outdated TensorFlow 1.5 source code. In addition to predicting position and orientation, for our use case, we must also predict the θ_1 and θ_2 angles for the two joints. While their approach involved training separate networks for position and orientation — effectively doubling computational requirements — we chose to train a single network that predicts all targets simultaneously. Additionally, Gessert et al. reported inefficiencies in their ResNext3D architecture due to the lack of grouped convolutions. We have incorporated grouped convolutions in our architecture to address this inefficiency.

3.2 Moon et al.'s V2V-PoseNet

Gessert et al. directly regressed the pose parameters. In this section, we describe an alternative which involves predicting keypoints on the tool from which the pose parameters — position, orientation, and joint angles — can be derived. The proposed arrangement of these keypoints is shown in Figure 4. From these keypoints, we can compute the necessary pose parameters as follows:

$$\begin{aligned}
\text{position} &= \frac{k_0 + k_1}{2} \\
\text{orientation} &= k_1 - k_0 \\
\theta_1 &= \text{angle_between_vectors}(k_1 - k_0, k_3 - k_2) \\
\theta_2 &= \text{angle_between_vectors}(k_3 - k_2, k_5 - k_4)
\end{aligned}$$

Here, k_i represents the 3D coordinates of keypoint i, and angle_between_vectors is a function that computes the angle between two vectors.

In direct regression, loss is typically calculated using mean squared error, which measures the squared difference between the predicted and target pose parameters.

Similarly, in keypoint estimation, mean squared error can be used to compute the difference between the predicted and target keypoint coordinates. However, a more effective approach for keypoint estimation is heatmap loss, where each keypoint's location is represented as a Gaussian heatmap centered on the target coordinate. An example of a heatmap can be seen in Figure 8. The model will then learn to predict heatmaps for each keypoint, and the loss will be calculated by comparing the predicted and target heatmaps using mean squared error loss. The keypoint's coordinate can be extracted from the predicted heatmap by applying the Argmax operation.

While directly regressing the pose is a straightforward solution, we argue that estimating keypoints with heatmap loss presents an advantage - stronger supervision signal. The model is required to predict a full spatial probability distribution for each keypoint rather than a single value as in direct regression. This leads to a detailed spatial gradient, which helps the model to converge faster and generalize better. This is particularly valuable when working with limited real-world data. Moreover, as stated by Moon et al. [6], directly regressing the pose requires the model to learn highly non-linear relationships. We investigate and prove this claim via multiple experiments, which are provided in the respective section.

3.2.1 Voxel-to-Voxel Prediction Network for Accurate 3D Hand Estimation

Moon et al. [6] proposed a novel method to estimate human hand keypoints from a single depth map. They argued that although depth maps inherently contain 3D information, many previous studies treat them as 2D data. Projecting 3D data into 2D introduces perspective distortions, making it more challenging for models to learn accurate representations. Furthermore, they noted that directly regressing keypoints from 2D data involves highly non-linear mappings, which further complicates the learning process.

To address these issues, Moon et al. converted the depth map into a 3D voxelized volume, allowing the network to operate on undistorted 3D data. Additionally, instead of direct regression, they estimated per-voxel likelihoods for each keypoint. For this task, they designed a custom network architecture called V2V-PoseNet. The network is based on the Hourglass Network [38] explained in the Background section. V2V-PoseNet takes a 3D voxel volume of size 64×64×64 as input and outputs a tensor of size Kx64x64x64 where K represents the number of keypoints. This way, for each keypoint, the network predicts a 3D volume, where the voxel values correspond to the likelihood of the keypoint being located at that coordinate. As explained before, to determine the coordinates of keypoint N, an Argmax operation is applied to the predicted volume at index N.

Although Moon et al.'s underlying data was a depth map image, they transformed it into voxel volumes for model training. While their study focused on hand keypoint estimation, their method is transferable to other objects, such as the surgical tool. These factors make V2V-PoseNet a strong baseline candidate for our study.

3.2.2 Out-of-View Estimation

The current method estimates the likelihood of keypoint being at each voxel. This presents an important limitation, as it restricts the model to predicting keypoints only within the volume's boundaries. Consequently, the method fails when keypoints fall outside the volume, such as when the tool is partially out of view.

Several approaches exist to address this issue, including padding the volume with empty voxels or artificially shifting out-of-view keypoints into the volume. In our approach, we tackle this problem by adjusting the model's output using simple shifting and scaling.

To illustrate this adjustment on a simple example, consider a scenario where the network predicts coordinates within the range $[0, 100]$, which corresponds to in-view keypoints, while the actual ground truth coordinates span $[-25, 125]$. First, we shift the ground truth coordinates by $+25$, changing the range to $[0, 150]$. Next, we scale the ground truth coordinates by a factor of $2/3$, bringing the range down to $[0, 100]$. The network is then trained to predict coordinates within this adjusted range. To restore the original span, the output coordinates are first scaled by $3/2$, bringing the range to $[0, 150]$, and then shifted by -25, returning them to the original range of $[-25, 125]$. This simple label pre-processing and output post-processing technique allows us to account for out-of-view keypoints with minimal loss in precision, which we consider acceptable.

3.2.3 Integral Regression

In traditional heatmap-based approaches, keypoint locations J_k are obtained by finding the position p with the maximum likelihood from the heatmap $H_k(p)$. This is formally represented as:

$$J_k = \arg\max_p H_k(p)$$

However, this approach has two main drawbacks: non-differentiability and quantization error. The $\arg\max$ operation above is non-differentiable, thereby reducing it to a post-processing step. This breaks the end-to-end nature of training, as supervision can only be imposed on the heatmaps, but not on the final joint coordinates. Quantization error arises because keypoints in real-world physical space have continuous coordinates, but when mapped to a discrete 3D voxel volume, these coordinates must be rounded to the nearest voxel, which leads to precision loss. While increasing the resolution of the voxel volume and the heatmap could reduce quantization error, it would come at the cost of higher computational and memory requirements.

Regression methods, in contrast, have two key advantages over heatmap-based methods: end-to-end learning and continuous output. Regression methods allow for end-to-end learning with direct supervision on the predicted keypoints. The predictions are

continuous and allow for precise localization, in principle, overcoming the quantization issues in heatmap-based methods.

To bridge the gap between these two methods, Sun et al. [49] introduced a unified approach that combines the strengths of both. Rather than applying the non-differentiable $\arg\max$, they proposed using an expectation over the heatmap to determine keypoint coordinates:

$$J_k = \sum_{p \in \Omega} p \cdot \tilde{H}_k(p)$$

where $\tilde{H}_k(p)$ is the normalized heatmap over the domain Ω. The normalization ensures that all values in the heatmap are non-negative and sum to 1. Specifically, the normalization is performed using the softmax function:

$$\tilde{H}_k(p) = \frac{\exp(H_k(p))}{\sum_{q \in \Omega} \exp(H_k(q))}$$

In this form, the keypoint J_k is obtained as the expected position in the domain Ω, weighted by the normalized heatmap probabilities. This can be interpreted as calculating the "center of mass" of the heatmap, where the probabilities serve as weights that determine the contribution of each voxel to the final coordinate. By transitioning from a non-differentiable $\arg\max$ operation to this differentiable expectation-based approach, the method enables end-to-end learning.

In practice, the discrete form of the expectation can be calculated as:

$$J_k = \sum_{z=1}^{D} \sum_{y=1}^{H} \sum_{x=1}^{W} p \cdot \tilde{H}_k(p)$$

where D, H, and W denote the resolution of the heatmap in depth, height, and width, respectively.

This method, often referred to as SoftArgmax, effectively combines the strengths of both heatmap-based and regression-based approaches, achieving both end-to-end learning and continuous joint localization with high accuracy.

3.3 Data Acquisition

3.3.1 Synthetic Data Generation

Training on real data generally yields better results since it more accurately reflects the complexities of real-world scenarios. However, obtaining real data can be costly and time-consuming, and in our case, the availability of real data was limited. To address this, we generated synthetic data and pre-trained our models on it to assess whether it could improve performance. Synthetic data offers a controlled and less complex environment

with sufficient sample quantity, making it easier to test models and debug issues before transitioning to more challenging real-world data.

In our synthetic data generation approach, we take advantage of the fact that we have a CAD model of our microsurgical tool, designed by the author's colleagues [19]. By using SolidWorks software, we can export the CAD model into STL format — a file type commonly used in 3D printing and computer-aided design that encodes information about 3D models. This STL file can then be converted into a voxel volume, which forms the basis of our synthetic dataset. Next, we outline the process of converting STL files into 3D voxel volumes, followed by a detailed explanation of the complete synthetic data generation pipeline that we utilize.

An STL file represents a 3D object as a triangular mesh, with its surface geometry composed of interconnected triangles (or facets). These triangles approximate the surfaces of the model. Typically, an STL file contains the vertices and normals of these triangles, but it lacks additional information, such as color, texture, and topological details, like the connectivity between triangles.

A straightforward method to convert an STL file into a voxel volume involves extracting the vertices from the mesh to create a point cloud. Then, by placing a voxel at each point in the cloud, the point cloud can be transformed into a voxel grid. However, this approach discards the mesh surface information, leading to data loss and sparse volumes. A better approach would involve incorporating the mesh information. As mentioned, an STL file defines the surface geometry of a 3D object using a collection of triangular polygons, with each triangle specified by its three vertices in 3D space. To convert this mesh into a 3D voxel grid, we need to determine whether each voxel lies inside or outside the object.

One way to determine this is to calculate the winding number, which measures how many times the surface of the mesh wraps around a point in space. Points inside the object will have a non-zero winding number, while points outside will have a winding number of zero. The winding number relative to a line segment (between two vertices of the mesh) can be calculated using the following formula:

$$\theta = \arctan 2(\text{end} - \text{pt}) - \arctan 2(\text{start} - \text{pt}) \tag{1}$$

This calculation must be performed for all line segments in the polygonal mesh. The sum of these angles provides the winding number at the voxel's position. If the winding number exceeds a certain threshold (typically π), the voxel is considered inside the object. Since implementing an efficient and accurate winding number calculation is complex and beyond the scope of this research, we utilized an open-source implementation [41] to handle this process.

Now that we can convert an STL file into a voxel volume, we describe our complete pipeline for synthetic data generation. First, we generate STL files representing every possible configuration of the microsurgical tool. The configuration is determined by

θ_1 and θ_2 joint angles. From this set, we randomly select a configuration, convert it into a voxel volume, and apply random translation and rotation. Finally, we introduce additional preprocessing steps, such as adding random objects and noise, as well as simulating uneven voxel spacing and signal loss to enhance realism.

Converting a single STL to a voxel volume is insufficient because the tool can have multiple configurations: the opening joint can rotate from 0 to 90 degrees, while the bending joint can move between -90 and 90 degrees. To account for all these possibilities, we developed a SolidWorks script that generates every configuration, resulting in 181 $\times$ 91 = 16,471 STL files. We then randomly select an STL file, convert it into a voxel volume, and apply further transformations.

In our use case, the microsurgical tool typically occupies approximately 70% of the actual 3D volume captured by the OCT device. To account for this, we first create an empty 3D volume of the required size. Next, we generate the voxel volume of the tool and paste it into the empty volume at a random position, simulating random translation. This method also allows control over how much of the tool is partially out of view. For random rotation, we apply rotation to the STL mesh before converting it to a voxel volume, as rotating a mesh is easier than rotating a voxel grid. To simulate the presence of random objects in the scene, we introduce parallelepipeds of varying sizes, placing them at random positions within the volume. Since STL files do not contain color information, converting them to 3D voxel volumes results in binary (black and white) volumes. However, due to the physical properties of OCT, surfaces closer to the device appear brighter in the resulting scan. To mimic this signal attenuation, we preprocess the synthetic data by gradually reducing the voxel intensity based on their depth.

Additionally, it is crucial to consider that OCT volumes typically exhibit uneven voxel spacing. This means that the physical distance between neighboring voxels along different axes (X, Y, and Z) is not necessarily the same. This stems from the way OCT devices acquire data, particularly due to variations in the optical resolution in different directions. When the volume is visualized, this discrepancy in voxel dimensions becomes apparent as a form of stretching or distortion along one or more axes. The specific voxel spacing is determined by the OCT device's design and acquisition settings, and it can vary between different systems. To account for this uneven voxel spacing in our synthetic data generation, we rescale the generated volume to match the required dimensions, artificially stretching or compressing it along specific axes to reflect the actual voxel spacing used in real OCT volumes.

Lastly, we add Gaussian and speckle noise to the volume. It is worth noting that OCT volumes contain a variety of noise types, including speckle noise, shot (photon) noise, thermal noise, quantization noise, and electronic interference noise, among others. However, accurately simulating all of these noise types is beyond the scope of this work. Therefore, for simplicity and practicality, we focus on adding Gaussian and speckle noise, which are the most prominent and relevant for our application.

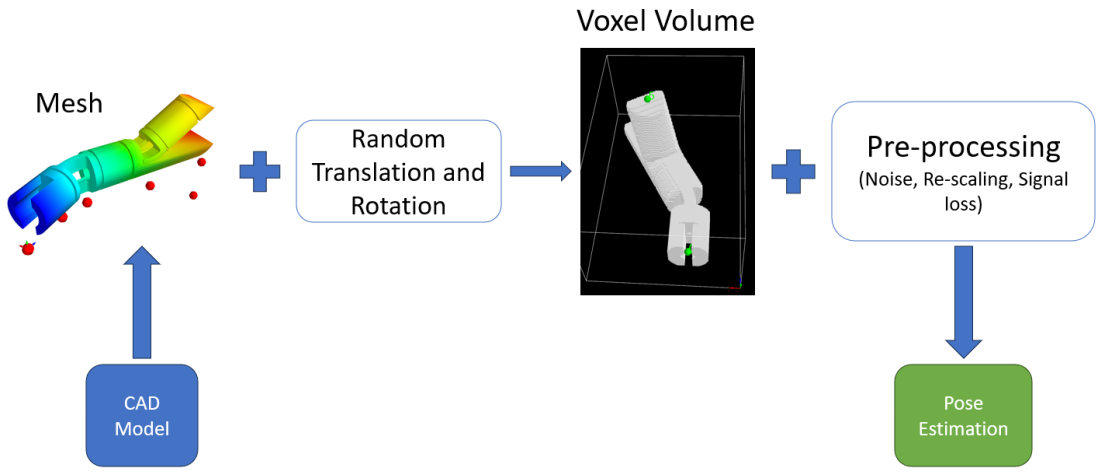

Figure 5. Synthetic Data Generation Pipeline. The CAD model of the surgical tool is converted into a mesh. Random translation and rotation are applied to the mesh, followed by conversion into a voxel volume. The voxel data is then pre-processed with noise addition, re-scaling, and simulation of signal loss to mimic real-world conditions. The resulting data is used for training the pose estimation model.

The complete synthetic data generation pipeline is illustrated in Figure 5. Using this pipeline, we have generated over 100,000 synthetic OCT volumes.

3.3.2 Real Data Collection and Annotation

Although conducting initial experiments on synthetic data can be beneficial, eventually, it becomes necessary to transition to using real data. At the time of this research, we have collected and annotated 192 real volumes. While we are continuing to gather and annotate more data, this study focuses on experiments conducted using these 192 volumes. The tool's position within these volumes varies with random translations and rotations. However, there are no additional objects present in the scene, and none of the volumes contain instances where the tool is partially out of view.

As previously mentioned, our OCT system is capable of capturing volumes with dimensions 1096x1936x1152, while the real-time version collects volumes at a reduced size of 28x1936x1152. Since our use case relies on the real-time system, it initially seemed practical to annotate the 28x1936x1152 volumes. However, we quickly discovered that annotating 28-slice volumes is more challenging, with the resulting annotations proving less accurate and consistent. We thus focused on annotating full-scale volumes and downsample them appropriately during model training.

We shall now discuss the approaches we used to annotate data. As discussed earlier, we decided to use models that either directly regress the targets or predict keypoints

from which the targets could be calculated. Thus, it made sense to focus on annotating keypoints, as targets could easily be derived from them. Although we do not plan to use models that predict bounding boxes, future studies could benefit from having such annotations available.

With this in mind, we explored various tools that could assist in data annotation. Several tools are available for annotating point clouds and biomedical volumes, such as CT or MRI scans. Unfortunately, tools designed for point clouds did not scale to our use case, and those developed for CT or MRI scans were primarily optimized for segmentation tasks. Unable to find an existing tool that met our specific needs, we decided to develop a custom annotation tool tailored to our application.

We leveraged Mayavi, a Python library for 3D scientific data visualization, which provides an intuitive way to visualize 3D voxel volumes and programmatically plot other geometries, such as points or parallelepipeds. Mayavi can be integrated into Qt applications, a popular framework for building graphical user interfaces (GUIs). This integration allowed us to embed Mayavi visualizations into a Qt application and modify them in real-time based on user interactions, such as button clicks. Using this setup, we developed an application that visualizes the given volume and allows users to input the x, y, z coordinates for each keypoint. Upon any user action (e.g. modifying coordinate x for keypoint K), the keypoints are re-rendered on the volume in real-time, without needing the user to re-open the application. Upon clicking the 'Save' button, the application generates an annotation file containing the volume's location on the disk and the keypoint coordinates.

This custom annotation tool enables us to annotate keypoints from which targets can be derived. Additionally, keypoints are placed in a way that makes it possible to derive bounding box positions. Another potential annotation option is to directly place bounding boxes, from which keypoints—and therefore targets—can be calculated. Visualizations of both annotation approaches can be seen in Figure 6. For the first approach, lines are drawn between specific pairs of keypoints to simplify the annotation process.

At this stage, it is unclear which annotation method provides more accurate and consistent results. To investigate this, we conducted an experiment in which three individuals (the author and his colleagues) annotated the same 30 volumes, resulting in three annotations per volume. For each volume, we then calculated the absolute average deviation between the three annotations. The final error was calculated as the mean of these absolute average deviations.

$$
\begin{aligned}
\text{mean} &= (annotation_1 + annotation_2 + annotation_3)/3 \\
\text{aed}_n &= |annotation_n - \text{mean}| \quad \text{(absolute average deviation)} \\
\text{error} &= (\text{aed}_1 + \text{aed}_2 + \text{aed}_3)/3
\end{aligned}
$$

Table 1 displays the mean error for each target. It is essential to emphasize that

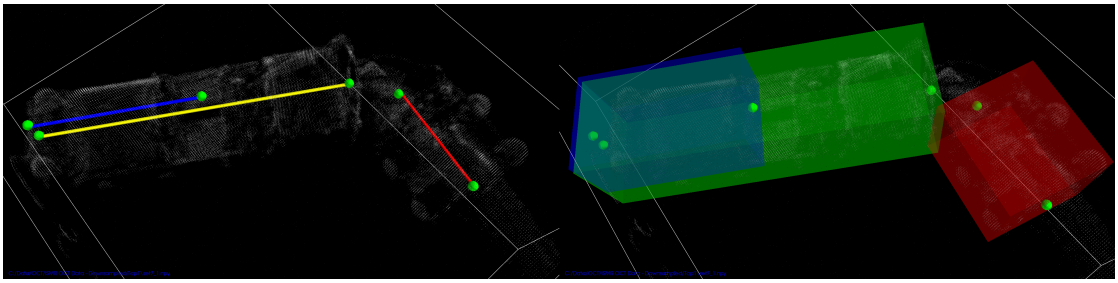

Figure 6. Annotation Approaches. The left panel shows keypoint-based annotation, where green dots are keypoints and connecting colored lines represent the links of the surgical tool. The right panel illustrates bounding box annotation, where overlapping boxes (blue, green, and red) define the spatial regions corresponding to the links of the tool. Both methods provide structured data for training the pose estimation model.

these errors stem from human annotations, and thus, the model's performance cannot be expected to surpass this level of accuracy. These errors represent the upper bound of the model's potential performance. Notably, the bounding box annotation approach resulted in smaller errors for position and θ_1, while the keypoint annotation approach yielded smaller errors for orientation and θ_2. Since position error is arguably more important for our use case, we decided to use the bounding box annotation approach to annotate the rest of the dataset.

Table 1. Comparison of annotation approaches for pose estimation. The table evaluates the accuracy of keypoint-based and bounding box-based annotations in terms of position error (in micrometers), orientation error (in degrees), and joint angle errors (θ_1 and θ_2, in degrees).

	Keypoints	**Bounding Boxes**
Position Error (µm)	89.7	62.7
Orientation Error (°)	0.36	0.42
θ_1 Error (°)	0.54	0.41
θ_2 Error (°)	0.22	0.27

3.4 Writing Assistance

Machine learning-based tools were used to improve the clarity and quality of this thesis. Specifically, ChatGPT [39], powered by the GPT-4 model, assisted with paraphrasing, sentence restructuring, error correction, and enhancing overall fluency while ensuring adherence to academic writing standards.

4 Experiments and Results

4.1 Experimental Setup

In this section, we train and evaluate three models discussed in the previous section: Gessert et al.'s Inception3D, Gessert et al.'s ResNext3D, and Moon et al.'s V2V-PoseNet. All the models have been trained from scratch without pre-training on synthetic data unless stated otherwise. The results are reported for four key metrics: position, orientation, θ_1, and θ_2 errors. Position errors are reported in millimeters, while orientation and joint errors are reported in degrees.

The dataset is divided into 80% for training (152 volumes) and 20% for validation (40 volumes). All final results are reported on the validation set. For experiments, we have used synthetic and real data described in the previous section. For Gessert et al.'s Inception3D and ResNext3D, which use direct regression, the targets were normalized to the range [0, 1].

The original input volumes (1096x1152x1936) are too large to fit in GPU memory, so we downsample them to reduce their size. Specifically, we take every 14th slice along all dimensions, resulting in a smaller volume of 78x82x138. For real-time use, the OCT scanner outputs volumes with only 28 slices in the first dimension. To match this, we further downsample the first dimension by taking every 3rd slice, resulting in an input volume of 28x82x138. This downsampling enables the following inference speeds: Inception3D achieves 18 ms per volume, V2V-PoseNet 20 ms, and ResNeXt3D 24 ms. Given our target inference speed of 20 Hz (50 ms per volume), all models meet the real-time requirement for our use case.

While smaller downsampling rates (e.g., taking every 10th slice) could yield more accurate models while remaining within the 50 ms limit, we opted for the current downsampling rate. This decision allows additional time for future processing steps, such as control algorithms, that may follow pose estimation in our envisioned robotics setup. As such, we did not investigate the effects of smaller or larger downsampling rates further.

Additionally, the input dimensions must be divisible by 8 to align with the architecture of our models. Inception3D and ResNeXt3D use three downsampling layers, each reducing the spatial dimensions by half, followed by a fully connected layer for final processing. Similarly, V2V-PoseNet employs three downsampling and three upsampling layers, with skip connections that require matching intermediate feature map sizes. To meet these requirements, we pad the volumes with empty voxel slices, resulting in a final input size of 32x88x144. Finally, we normalize the voxel values to the range [0, 1].

To enhance the generalization capability of the models, random flipping along the y and z axes was applied to augment the training data. However, no regularization techniques, such as weight decay or dropout, were used, and no learning rate scheduler was applied. The rationale for not applying regularization is discussed in the following

section.

All models are trained for 500 epochs with a batch size of 25. For Gessert et al.'s Inception3D and ResNext3D models, the learning rate is set to 1e-4, while for Moon et al.'s V2V-PoseNet, it is set to 1e-5. In synthetic volumes, the surgical tool can be out of view. To account for this, a coordinate shifting and scaling adjustment has been applied for V2V-PoseNet, which allows up to 20% of the surgical tool to be out of view in the volume.

4.2 Effects of Model Pre-training

Given that we only have 192 annotated real volumes compared to 100,000 synthetic volumes, we applied transfer learning, explained in the Background section, to assess whether pre-training models on synthetic data and then fine-tuning them on real data yields better results than training models from scratch on real data alone. We have pre-trained Inception3D, ResNeXt3D, and V2V-PoseNet on synthetic data, then selected the best-performing checkpoint from each model and fine-tuned them on real data. Additionally, we trained all models on real data from scratch for comparison.

The results are shown in Table 2. For visualizations of the results, please see Figure 9, Figure 10 and Figure 11. The findings demonstrate that pre-training improves performance for Inception3D and ResNeXt3D. Position error decreased by around 1.5 and 0.3 millimeters, orientation error by 5 and 1 degree, θ_1 error by 3 and 2 degrees, and θ_2 error by 3 and 1 degrees respectively. Notably, although Inception3D and ResNeXt3D showed considerable improvement, V2V-PoseNet did not exhibit any gains. These findings suggest that pre-training Inception3D and ResNeXt3D on our synthetic data is an effective way to improve the performance on real-world data with few annotations. Moreover, the results hint that Inception3D and ResNeXt3D might require more training data compared to V2V-PoseNet.

After confirming that transfer learning enhances results, we focused on comparing the models' performance. Surprisingly, Inception3D performed the worst despite being the best-performing model in Gessert et al.'s study. Both ResNeXt3D and V2V-PoseNet performed at a similar level, with V2V-PoseNet demonstrating superior accuracy in position estimation, while ResNeXt3D performed slightly better in orientation and joint angle estimation accuracy. This aligns with our expectations since ResNeXt3D is directly optimized for orientation and joint angles, while V2V-PoseNet is not and computes these pose parameters during post-processing.

Given that we consider position error to be more critical than orientation error and joint angles errors, and the differences in these errors were relatively small, we decided to proceed with V2V-PoseNet for further experimentation. Figure 7 illustrates the keypoint predictions made by V2V-PoseNet for a sample from the validation set, highlighting the model's accuracy in predicting keypoints locations compared to the ground truth.

Model	Pos Err (mm)	Ornt Err (°)	θ_1 Err (°)	θ_2 Err (°)
Inception3D No pretraining	2.1 ± 0.8	7.9 ± 2.6	9.0 ± 7.9	7.0 ± 4.4
Inception3D Pretrained	0.61 ± 0.29	2.4 ± 2.0	5.6 ± 4.6	3.5 ± 2.5
ResNeXt3D No pretraining	0.83 ± 0.44	2.8 ± 3.2	4.1 ± 4.0	2.6 ± 2.8
ResNeXt3D Pretrained	0.48 ± 0.21	**2.1 ± 1.9**	**2.1 ± 1.6**	**1.1 ± 1.0**
V2V-PoseNet No pretraining	**0.36 ± 0.17**	2.9 ± 3.0	2.1 ± 2.1	2.6 ± 1.9
V2V-PoseNet Pretrained	0.38 ± 0.46	3.9 ± 7.4	3.5 ± 7.5	2.2 ± 1.4

Table 2. Comparison of the performance of Inception3D, ResNeXt3D, and V2V-PoseNet models with and without pre-training.

4.3 V2V-PoseNet: Keypoint Estimation vs. Direct Regression

We decided to conduct an ablation study to determine whether the performance of V2V-PoseNet is driven primarily by its keypoint estimation and heatmap loss, or if the architecture itself is responsible for the observed results. This would help us verify whether the keypoint estimation strategy gives the model its edge or if the architecture alone could achieve similar results with direct regression.

To test this, we replaced the final layers of V2V-PoseNet with three CNN blocks, each followed by 2x2x2 pooling. The tensor was then flattened and passed through a fully connected layer to output the same 8 targets as in Gessert et al.'s direct regression model. Additionally, we normalized the targets to the range [0, 1], mirroring the setup of Gessert et al.'s direct regression approach. We ensured that the modified network had minimal changes compared to the original V2V-PoseNet architecture and that both models had the same number of parameters to maintain fairness in terms of modeling capacity.

The results of our investigation, shown in Table 3 and in Figure 12, demonstrate that direct regression of position, orientation, and angles using the modified V2V-PoseNet architecture leads to considerable deterioration in performance. This aligns with our expectations as V2V-PoseNet (which, as a reminder, is based on Hourglass Network) was not designed for direct regression, and Inception3D (the worst performing model) has an arguably better architecture for such task. Thus, this experiment confirms that heatmap prediction and heatmap loss are crucial to achieving the model's strong performance.

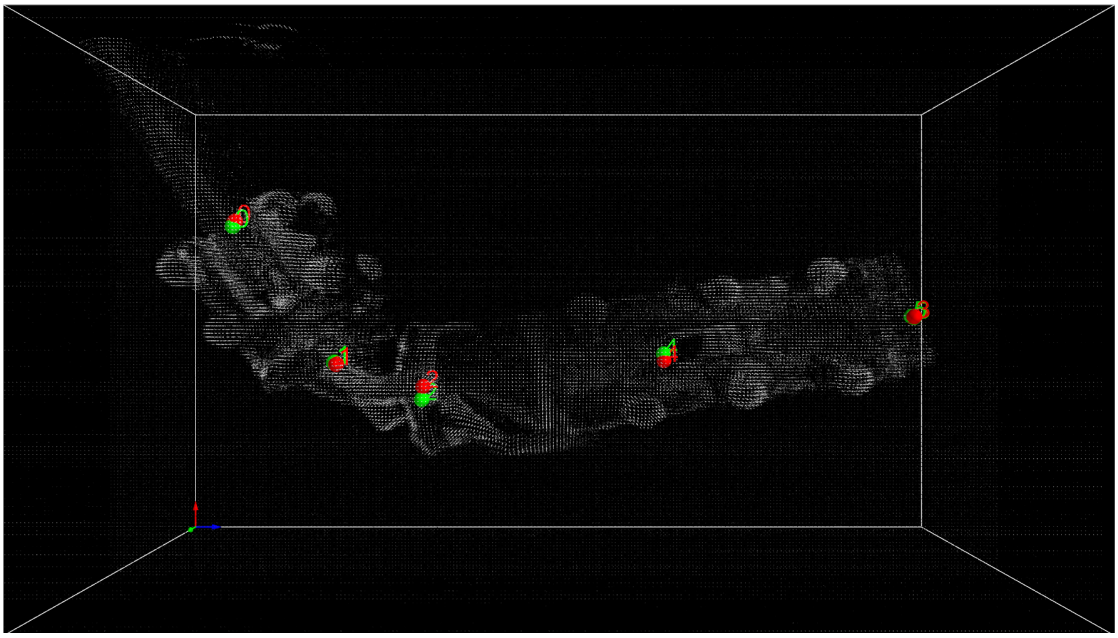

Figure 7. Keypoint predictions by V2V-PoseNet for a validation set volume. Green dots represent the ground truth keypoint locations, while red dots indicate the model's predicted keypoints.

4.4 V2V-PoseNet: Gaussian vs. One-Hot Heatmap Loss

Since we have established that heatmap prediction and heatmap loss notably contribute to V2V-PoseNet's performance, we decided to investigate whether a different heatmap generation method could yield better results. Currently, we use Gaussian heatmap generation, which creates a smooth, bell-shaped distribution around the keypoint. This assigns higher values to voxels closer to the keypoint and gradually decreases for those further away, making it easier for the model to focus on the regions near the keypoint.

In contrast, we explored an alternative approach using one-hot heatmap generation, which, despite its name, is not a true one-hot representation. Instead, it creates a probabilistic distribution over the nearest voxels to the keypoint. The voxels closest to the keypoint receive the highest probabilities, and these probabilities are interpolated based on the distance from the keypoint. This approach allows for some flexibility in the keypoint's position, as the probability distribution accounts for the keypoint's proximity to multiple voxels.

To illustrate the difference, consider a simple 1D example with a tensor of size 7 where the ground truth coordinate is 2.8. A Gaussian heatmap would generate a smooth distribution like "0.0, 0.6, 0.8, 1.0, 0.8, 0.6, 0.0," while the one-hot heatmap would produce a more concentrated distribution such as "0.0, 0.0, 0.1, 0.9, 0.0, 0.0, 0.0," where the probability is interpolated across the neighboring voxels. The visualization of both

Model	Pos Err (mm)	Ornt Err (°)	θ_1 Err (°)	θ_2 Err (°)
V2V-PoseNet Direct Regression	4.9 ± 1.1	9.1 ± 3.0	22.0 ± 15.3	13.2 ± 4.4
V2V-PoseNet No pretraining	0.36 ± 0.17	2.9 ± 3.0	2.1 ± 2.1	2.6 ± 1.9

Table 3. Comparison of V2V-PoseNet performance using direct regression versus keypoint estimation. The results highlight the considerable deterioration in performance when replacing keypoint estimation with direct regression.

heatmap approaches is presented in Figure 8.

As shown in Table 4 and Figure 13, our results indicate that Gaussian heatmaps perform considerably better. This might hint that compared to the one-hot approach, the Gaussian heatmap provides richer supervision, which enables the model to learn more robust spatial relationships around the keypoint. It is also possible that the smoothness of the Gaussian heatmap likely facilitates better gradient flow during training, making the optimization process more stable and effective.

Model	Pos Err (mm)	Ornt Err (°)	θ_1 Err (°)	θ_2 Err (°)
One-Hot Heatmap	0.9 ± 3.1	8.2 ± 28.0	17.8 ± 47.8	13.9 ± 37.5
Gaussian Heatmap	0.36 ± 0.17	2.9 ± 3.0	2.1 ± 2.1	2.6 ± 1.9

Table 4. Comparison of V2V-PoseNet performance using Gaussian heatmap generation versus one-hot heatmap generation. The results demonstrate that Gaussian heatmaps outperform one-hot heatmaps, suggesting that the richer supervision provided by Gaussian heatmaps enables better spatial learning.

4.5 V2V-PoseNet: SoftArgmax and Mixed Loss

A key limitation of the current V2V-PoseNet model is its reliance on the Argmax function, which is non-differentiable and cannot be used for end-to-end training. To address this, we replaced Argmax with Integral Regression, commonly referred to as SoftArgmax, a differentiable alternative. This modification enables the model to compute keypoint coordinates and pose parameters while allowing gradients to flow through the entire process. Consequently, the model can directly optimize for keypoint coordinates and pose parameters during training.

SoftArgmax has an important hyperparameter - temperature - that controls the sharpness of the probability distribution produced by the SoftMax function. A lower temperature produces a smoother distribution, while a higher temperature sharpens predictions, making them more 'confident'. To determine the optimal temperature, we conducted

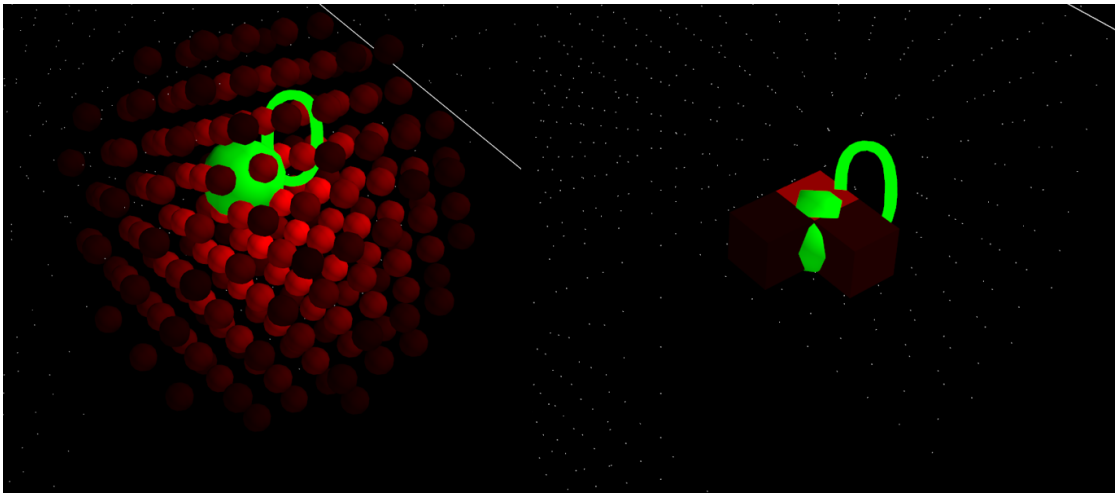

Figure 8. Visualization of different approaches for generating a heatmap for a keypoint. The green sphere represents the keypoint coordinate, while the red spheres illustrate the corresponding heatmap values, with brighter spheres indicating higher values. The left panel displays a Gaussian-based heatmap, showing smoothly distributed intensity around the keypoint. The right panel depicts a one-hot-based heatmap, with intensity sharply concentrated near the keypoint.

experiments with various values. At temperatures of 1 and 10, the model struggled to learn accurate coordinates. Upon further investigation, we hypothesize that at these lower temperatures, the predicted heatmaps do not have sufficient contrast between the peak (true keypoint) and surrounding voxels. This causes the Softmax function to produce distributions that are too smooth, resulting in incorrect "center of mass" calculations and inaccurate coordinates. When the temperature was increased to 1000, the distribution became overly sharp, approaching a near one-hot representation. This caused gradient flow issues and often led to vanishing gradients during backpropagation. Consequently, the model's ability to learn was impaired. A temperature of 100 provided the best balance. It was high enough to separate the true peak from surrounding voxels, enabling accurate coordinate calculation, yet low enough to avoid vanishing gradient problems.

Since SoftArgmax allows the prediction of keypoint coordinates and pose parameters, we also experimented with assigning different importance to these components by adjusting loss function weights. For example, we prioritized position error over joint angle errors in some configurations. However, our results showed no benefit to this approach. In fact, the model performed best when relying solely on heatmap and keypoint losses without additional weighting modifications.

Finally, we explored different training strategies, such as training with both heatmap and coordinate losses simultaneously or using a sequential approach — training with heatmap loss only first, followed by coordinate loss only. Our experiments revealed

no advantage to the sequential approach; it generally produced slightly worse results compared to simultaneous training of both loss components.

Table 5 and Figure 14 summarize the performance of V2V-PoseNet with Integral Regression (SoftArgmax) across different temperature values. For these experiments, the heatmap and coordinate loss weights were set to 1, while other loss weights (e.g., position, orientation, and joint angles) were set to 0. The models were trained using simultaneous heatmap and coordinate losses. As explained earlier, the model with a temperature of 100 achieved the best results, even outperforming the original V2V-PoseNet. Specifically, the position error decreased by 0.1 millimeters, orientation error - by 0.6 degrees, θ_1 and θ_2 errors - by 0.2 and 0.6 degrees, respectively, which is a considerable improvement in performance.

Model	**Pos Err (mm)**	**Ornt Err (°)**	**θ_1 Err (°)**	**θ_2 Err (°)**
V2V-PoseNet No pretraining	0.36 ± 0.17	2.9 ± 3.0	2.1 ± 2.1	2.6 ± 1.9
SoftArgMax (T=1)	2.7 ± 0.6	30.5 ± 2.1	33.7 ± 17.1	18.6 ± 8.4
SoftArgMax (T=10)	1.1 ± 0.5	8.5 ± 5.9	9.1 ± 8.4	5.5 ± 5.6
SoftArgMax (T=100)	**0.26 ± 0.11**	**2.3 ± 2.1**	**1.9 ± 1.8**	**1.9 ± 1.5**
SoftArgMax (T=1000)	7.5 ± 2.7	24.4 ± 7.7	28.2 ± 12.5	11.2 ± 7.2

Table 5. Performance comparison of V2V-PoseNet with SoftArgmax across different temperature values. A temperature of 100 achieves the best balance between gradient flow and smoothness of probability distribution in SoftMax, resulting in performance that surpasses the original V2V-PoseNet.

4.6 V2V-PoseNet: Regularization Techniques

After analyzing the training and validation logs, we observed that V2V-PoseNet with SoftArgmax exhibited less overfitting compared to the original V2V-PoseNet. This suggests that the improved performance of the SoftArgmax-based model may be due to its reduced overfitting rather than the introduction of SoftArgmax itself. To investigate this hypothesis, we applied several regularization strategies to the original V2V-PoseNet, including Weight Decay, Dropout, Random Translation, Random Occlusions, and Mixup [59]. Five additional models were trained to evaluate the effectiveness of these strategies.

In the first approach, we applied weight decay to the model's parameters to mitigate overfitting by penalizing large weights. After experimenting with various values,

Model	**Pos Err (mm)**	**Ornt Err (°)**	**θ_1 Err (°)**	**θ_2 Err (°)**
V2V-PoseNet No pretraining	0.36 ± 0.17	2.9 ± 3.0	2.1 ± 2.1	2.6 ± 1.9
WeightDecay=0.0001	0.38 ± 0.19	3.3 ± 7.6	2.3 ± 2.6	**2.1 ± 1.9**
Dropout=0.05	**0.33 ± 0.15**	2.7 ± 3.0	2.0 ± 1.7	2.2 ± 1.6
Random Shift	0.34 ± 0.16	3.0 ± 2.6	2.5 ± 2.1	2.3 ± 1.8
Random Occlusion	0.35 ± 0.16	2.8 ± 2.4	2.1 ± 1.3	2.4 ± 1.9
Mixup	0.35 ± 0.18	**2.6 ± 2.1**	**1.7 ± 1.2**	2.4 ± 1.7

Table 6. Comparison of V2V-PoseNet performance under various regularization techniques. The results indicate that none of the techniques yielded a considerable improvement over the baseline performance.

we found that weight decay in the range of 1×10^{-4} to 1×10^{-5} provided the best performance.

In the second approach, we added dropout at various layers of the network to randomly set some activations to zero during training, which prevents the model from over-relying on any one node. In convolutional architectures, it is common to add dropout after the fully connected layers or after the final convolutional layers. We experimented with dropout rates between 0.05 and 0.2, as these are reasonable values for reducing overfitting without drastically impairing the model's learning capacity.

In the third approach, we applied random translations to the input volumes by shifting the data by a randomly selected number of slices in random directions (left, right, top, bottom, front, back). After shifting, the empty space left behind can either be left as zeros, filled with noise, or filled by mirroring the part that was shifted out of view. Our experiments showed that filling the empty space with noise provided the best results.

In the fourth approach, we introduced random occlusions by inserting parallelepipeds of random sizes into the volumes, following a similar approach used during synthetic data generation. This technique helps the model learn to handle missing or occluded data, making it more robust.

In the fifth approach, we leveraged Mixup as a data augmentation technique. Mixup generates synthetic training examples by linearly interpolating between two training examples and their labels. Specifically, given two input volumes x_i and x_j and their corresponding heatmaps h_i and h_j, Mixup creates a new sample as follows:

$$\begin{aligned}\tilde{x} &= \lambda x_i + (1-\lambda)x_j \\ \tilde{y} &= \lambda h_i + (1-\lambda)h_j \\ \lambda &\sim \text{Beta}(\alpha, \alpha)\end{aligned}$$

where λ is drawn from a Beta distribution with a parameter α. In our case, we used

$\alpha = 0.2$, which controls the degree of interpolation between the examples. Mixup has been shown to improve generalization by encouraging the model to behave linearly between training examples, reducing overfitting.

Our experimental results, shown in Table 6 and in Figure 15, indicate that none of these regularization techniques improved the validation scores. We attribute this to the limited amount of available annotated real data and believe that larger datasets may naturally help improve the model's performance.

5 Conclusion

Accurately tracking the pose of multi-jointed microsurgical tools in real-time is a critical challenge in minimally invasive and robotic-assisted surgery, particularly for delicate procedures like neurosurgery. This thesis addressed this challenge by developing a markerless, high-speed, and accurate pose estimation method for an 8-degree-of-freedom microsurgical tool using optical coherence tomography. The proposed method achieves an average position error of 0.26 millimeters, orientation error of 2.3 degrees, and joint angle errors of 1.9 and 1.9 degrees for θ_1 and θ_2, respectively, while operating with an inference speed of 20 milliseconds per volume. These results demonstrate the efficacy of our approach in providing reliable pose estimation for dynamic surgical environments. Our solution leverages deep learning techniques to process volumetric OCT data, offering robust performance even in scenarios where portions of the tool are occluded or out of view. Additionally, by adopting a markerless approach, we eliminate the need for physical markers, which are prone to occlusion and may interfere with magnetic actuation.

Despite these achievements, certain limitations remain. A primary constraint is the limited availability of annotated real-world OCT data, which necessitated extensive use of synthetic data for model pre-training. Future studies should also prioritize testing the proposed method in a real in-vivo surgical setup to assess its practical applicability and uncover potential challenges. Lastly, exploring alternative data representations, such as point clouds or multi-view approaches like Virtual Viewpoint Selection by Jian Cheng and Yanguang Wan [7], could prove worthwhile. These methods, which extract features from multiple 2D perspectives or leverage 3D point data, may offer a more computationally efficient and potentially more accurate alternative to voxel-based processing, providing further advancements in real-time pose estimation.

6 Contributions

This section clarifies the author's individual contributions to this thesis, distinguishing personal work from collaborative efforts within the joint research project.

The author was primarily responsible for the development of the deep learning methods used for real-time markerless pose estimation of a microsurgical tool from OCT volumes. This included re-implementing the Inception3D and ResNeXt3D architectures originally proposed by Gessert et al., with modifications such as the addition of grouped convolutions for improved computational efficiency. The author also adapted the V2V-PoseNet architecture by Moon et al. and incorporated Integral Regression (SoftArgmax), introduced by Sun et al., as a differentiable alternative to the Argmax function. Several heatmap generation strategies, including Gaussian and One-Hot heatmaps, were explored and implemented to assess their impact on model performance.

The author also developed a synthetic data generation pipeline using STL files of the microsurgical tool, which involved converting STL meshes into voxel volumes and applying various preprocessing steps to simulate real-world conditions. Additionally, the author created a custom annotation tool using Mayavi and Qt to streamline the process of labeling real OCT volumes. All the experiments presented in this work, along with the analysis and interpretation of results, were designed and conducted by the author. The author was also fully responsible for writing this thesis document.

In the context of this thesis, the author's colleagues were responsible for the design and fabrication of the microsurgical tool, OCT data acquisition, and the envisioned robotic setup for neurosurgery, among other significant contributions not detailed in this work. They also handled the conversion of the CAD model to STL format and the generation of all possible tool configurations, which was essential for synthetic data generation.

The collaborative aspects of this project, among others, include data annotation, which was performed equally by all three collaborators, the review of existing literature, as well as regular discussions on intermediate results and potential next steps. Throughout the project, the author benefited greatly from the valuable guidance, feedback, and advice provided by the author's colleagues.

7 Acknowledgments

The author wishes to express deepest gratitude to the supervisors, Dr. Dmytro Fishman and Dr. Lueder Kahrs, for their invaluable guidance, insights, and support throughout this journey. Their expertise and encouragement have been essential for this project.

The author is also deeply grateful to his teammates, Nirmal Pol and Erik Fredin, whose support and contributions have also been integral to this project. Working together has been an incredible experience.

Lastly, the author extends heartfelt thanks to the University of Toronto for organizing "Summer Program for Students from Ukraine." This program provided Ukrainians with an opportunity to work with and learn from outstanding researchers in the field. Through this program, the author had the privilege of working in the Medical Computer Vision and Robotics Lab under the supervision of Dr. Lueder Kahrs, which laid the foundation for this project.

References

[1] Parikshith Reddy Baddam. Surgical robotics unveiled: The robotic surgeon's role in modern surgical evolution. *ABC Journal of Advanced Research*, 2019.

[2] S. Bhowmik, C. B. Singh, S. Neogi, and S. Roy. Evaluating the emergency surgery score (ess) in predicting postoperative outcomes following emergency laparotomy: Insights from an indian tertiary center. *Cureus*, 16(3):e56455, Mar 2024.

[3] David Bouget, Max Allan, Danail Stoyanov, and Pierre Jannin. Vision-based and marker-less surgical tool detection and tracking: a review of the literature. *Medical Image Analysis*, 35:633–654, 2017.

[4] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1):172–186, 2021.

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Computer Vision – ECCV 2020*, pages 213–229, Cham, 2020. Springer International Publishing.

[6] Ju Yong Chang, Gyeongsik Moon, and Kyoung Mu Lee. V2v-posenet: Voxel-to-voxel prediction network for accurate 3d hand and human pose estimation from a single depth map. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5079–5088, 2018.

[7] Jian Cheng, Yanguang Wan, Dexin Zuo, Cuixia Ma, Jian Gu, Ping Tan, Hongan Wang, Xiaoming Deng, and Yinda Zhang. Efficient virtual view selection for 3d hand pose estimation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(1):419–426, Jun. 2022.

[8] P. J. Choi, R. J. Oskouian, and R. S. Tubbs. Telesurgery: Past, present, and future. *Cureus*, 10(5):e2716, May 2018.

[9] Ricardo Coletta, Tuula Salo, and Andrew Yeudall. Current trends on prevalence, risk factors and prevention of oral cancer. *Frontiers in Oral Health*, 5:1505833, Nov 2024.

[10] Giulio Dagnino and Dennis Kundrat. Robot-assistive minimally invasive surgery: Trends and future directions. *International Journal of Intelligent Robotics and Applications*, 8(4):812–826, Dec 2024.

[11] Maria Dalamagka. Pathology associated with kidney failure and anesthesia. *Magna Scientia Advanced Research and Reviews*, 12:106–109, 10 2024.

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ArXiv*, abs/2010.11929, 2020.

[13] Panagiotis Doukas, Oliver Hartmann, Jelle Frankort, Birte Arlt, Hanif Krabbe, Michael Johan Jacobs, Andreas Greiner, Jan Paul Frese, and Alexander Gombert. Postoperative bioactive adrenomedullin is associated with the onset of ards and adverse outcomes in patients undergoing open thoracoabdominal aortic surgery. *Scientific Reports*, 14(1):12795, 2024.

[14] Justis Ehlers, Sunil Srivastava, and Daniel Feiler et al. Integrative advances for oct-guided ophthalmic surgery and intraoperative oct: Microscope integration, surgical instrumentation, and heads-up display surgeon feedback. *PLoS ONE*, 9(8):e105224, Aug 2014.

[15] Robert Elfring, Matías de la Fuente, and Klaus Radermacher. Assessment of optical localizer accuracy for computer aided surgery systems. *Computer Aided Surgery*, 15(1-3):1–12, 2010. PMID: 20233129.

[16] Nima Enayati, Elena De Momi, and Giancarlo Ferrigno. Haptics in robot-assisted surgery: Challenges and benefits. *IEEE Reviews in Biomedical Engineering*, 9:49–65, 2016.

[17] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press, 1996.

[18] Markus Finke, Sven Kantelhardt, and Alexander Schlaefer et al. Automatic scanning of large tissue areas in neurosurgery using optical coherence tomography. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 8(3):327–336, Sep 2012.

[19] Cameron Forbrigger, Erik Fredin, and Eric Diller. Evaluating the feasibility of magnetic tools for the minimum dynamic requirements of microneurosurgery. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4703–4709, 2023.

[20] Erik Fredin, Nirmal Pol, Anton Zaliznyi, Eric Diller, and Lueder A. Kahrs. Estimating the joint angles of an articulated microrobotic instrument using optical coherence tomography. In *2024 International Symposium on Medical Robotics (ISMR)*, pages 1–7, 2024.

[21] Corey K. Gentle, Moustafa Moussally, Jenny H. Chang, Hanna Hong, Kelly Walker, Kelly Nimylowycz, Sayf Al-deen Said, and Zahraa Al-Hilli. Beyond cdc-defined surgical site infection: Factors associated with antibiotic prescription after breast operation. *Surgical Infections*, 0(0):null, 0. PMID: 39504129.

[22] Nils Gessert, Matthias Schlüter, and Alexander Schlaefer. A deep learning approach for pose estimation from volumetric oct data. *Medical Image Analysis*, 46:162–179, 2018.

[23] M. Guan, H. Li, T. Tian, J. Peng, Y. Huang, and L. He. Different minimally invasive surgical methods to hysterectomy for benign gynecological disease: A systematic review and network meta-analysis. *Health Science Reports*, 7(11):e70137, Nov 2024.

[24] Naomi Gutkind. Standalone migs in the current glaucoma treatment paradigm, 2024.

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[26] Yisheng He, Wei Sun, Haibin Huang, Jianran Liu, Haoqiang Fan, and Jian Sun. Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[27] P. Healey and J. Samanta. When does the 'learning curve' of innovative interventions become questionable practice? *European Journal of Vascular and Endovascular Surgery*, 36(3):253–257, 2008.

[28] Kristina Irsch. *Optical Principles of OCT*, pages 1–14. Springer International Publishing, Cham, 2020.

[29] Debesh Jha, Sharib Ali, Nikhil Kumar Tomar, Michael A. Riegler, Dag Johansen, Håvard D. Johansen, and Pål Halvorsen. Exploring deep learning methods for real-time surgical instrument segmentation in laparoscopy. In *2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*, pages 1–4, 2021.

[30] Julia Adriana Kasmirski et al. Where the rubber meets the road: Pearls and pitfalls of implementing competency-based assessment. *ANZ Journal of Surgery*, 94(11):1906–1909, Nov 2024.

[31] Florian Kral, Elisabeth Puschban, Herbert Riechelmann, and Wolfgang Freysinger. Comparison of optical and electromagnetic tracking for navigated lateral skull base surgery. *The International Journal of Medical Robotics + Computer Assisted Surgery*, 9(2):247–252, 2013.

[32] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[33] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[34] Dongfang Li, Dingran Dong, Wahshing Lam, Liuxi Xing, Tanyong Wei, and Dong Sun. Automated in vivo navigation of magnetic-driven microrobots using oct imaging feedback. *IEEE Transactions on Biomedical Engineering*, 67(8):2349–2358, 2020.

[35] Shuying Liu and Weihong Deng. Very deep convolutional neural network based image classification using small training sample size. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 730–734, 2015.

[36] Cristian Mornos and Adrian-Sebastian Zus. *Renal Vascular Anomalies*, pages 31–43. Springer Nature Switzerland, Cham, 2024.

[37] Peter Mountney, Razvan Ionasec, Markus Kaizer, Sina Mamaghani, Wen Wu, Terrence Chen, Matthias John, Jan Boese, and Dorin Comaniciu. Ultrasound and fluoroscopic images fusion by autonomous ultrasound probe detection. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012*, pages 544–551, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[38] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *Computer Vision – ECCV 2016*, pages 483–499, Cham, 2016. Springer International Publishing.

[39] OpenAI. Chatgpt, 2023.

[40] Georgios Pavlakos, Xiaowei Zhou, Aaron Chan, Konstantinos G. Derpanis, and Kostas Daniilidis. 6-dof object pose from semantic keypoints. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2011–2018, 2017.

[41] Christian Pederkoff. stl-to-voxel, 2021.

[42] L. Qiu, C. Li, and H. Ren. Real-time surgical instrument tracking in robot-assisted surgery using multi-domain convolutional neural network. *Healthcare Technology Letters*, 6(6):159–164, Dec 2019.

[43] Shahzad Raja, Umberto Benedetto, and Nandor Marczin. Inflammation and heart surgery. *Frontiers in Cardiovascular Medicine*, 11:1493898, Sep 2024.

[44] Lars Richter, Peter Trillenberg, Achim Schweikard, and Alexander Schlaefer. Stimulus intensity for hand held and robotic transcranial magnetic stimulation. *Brain Stimulation: Basic, Translational, and Clinical Research in Neuromodulation*, 6(3):315–321, 2013.

[45] Y. Rivero-Moreno, M. Rodriguez, P. Losada-Muñoz, S. Redden, S. Lopez-Lezama, A. Vidal-Gallardo, D. Machado-Paled, J. Cordova Guilarte, and S. Teran-Quintero. Autonomous robotic surgery: Has the future arrived? *Cureus*, 16(1):e52243, Jan 2024.

[46] Matthias Schlüter, Lukas Glandorf, Martin Gromniak, Thore Saathoff, and Alexander Schlaefer. Concept for markerless 6d tracking employing volumetric optical coherence tomography. *Sensors*, 20(9), 2020.

[47] Nabil Simaan, Rashid Yasin, and Long Wang. Medical technologies and challenges of robot-assisted minimally invasive intervention and diagnostics. *Annual Review of Control, Robotics, and Autonomous Systems*, 1, 05 2018.

[48] Hao Su, Ka-Wai Kwok, Kevin Cleary, Iulian Iordachita, M. Cenk Cavusoglu, Jaydev P. Desai, and Gregory S. Fischer. State of the art and future opportunities in mri-guided robot-assisted surgery and interventions. *Proceedings of the IEEE*, 110(7):968–992, 2022.

[49] Xiao Sun, Bin Xiao, Shuang Liang, and Yichen Wei. Integral human pose regression, 11 2017.

[50] Xi Tang. Research progress on enteral nutrition in patients with esophageal cancer. *MedScien*, 1(9), Oct 2024.

[51] Yuankai K. Tao, Sunil K. Srivastava, and Justis P. Ehlers. Microscope-integrated intraoperative oct with electrically tunable focus and heads-up display for imaging of ophthalmic surgical maneuvers. *Biomed. Opt. Express*, 5(6):1877–1885, Jun 2014.

[52] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1653–1660, 2014.

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[54] Ina Vernikouskaya, Dagmar Bertsche, Wolfgang Rottbauer, and Volker Rasche. 3d-xguide: Open-source x-ray navigation guidance system. *International Journal of Computer Assisted Radiology and Surgery*, 16(1):53–63, 2021.

[55] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. Densefusion: 6d object pose estimation by iterative dense fusion. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3338–3347, 2019.

[56] Qianqian Wang, Lidong Yang, Jiangfan Yu, Chi-Ian Vong, Philip Wai Yan Chiu, and Li Zhang. Magnetic navigation of a rotating colloidal swarm using ultrasound images. *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5380–5385, 2018.

[57] Yan Wang, Qiyuan Sun, Zhenzhong Liu, and Lin Gu. Visual detection and tracking algorithms for minimally invasive surgical instruments: A comprehensive review of the state-of-the-art. *Robotics and Autonomous Systems*, 149:103945, 12 2021.

[58] Konstantin Yashin, Matteo Mario Bonsanto, Ksenia Achkasova, Anna Zolotova, Al-Madhaji Wael, Elena Kiseleva, Alexander Moiseev, Igor Medyanik, Leonid Kravets, Robert Huber, Ralf Brinkmann, and Natalia Gladkova. Oct-guided surgery for gliomas: Current concept and future perspectives. *Diagnostics*, 12(2), 2022.

[59] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. Mixup: Beyond empirical risk minimization. arXiv, 2017.

[60] Yaokun Zhang and Heinz Wörn. Optical coherence tomography as highly accurate optical tracking system. *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 1145–1150, 2014.

[61] Mingchuan Zhou, Xing Hao, Abouzar Eslami, Kai Huang, Caixia Cai, Chris P. Lohmann, Nassir Navab, Alois Knoll, and M. Ali Nasseri. 6dof needle pose estimation for robot-assisted vitreoretinal surgery. *IEEE Access*, 7:63113–63122, 2019.

Appendix

I. Result Visualizations

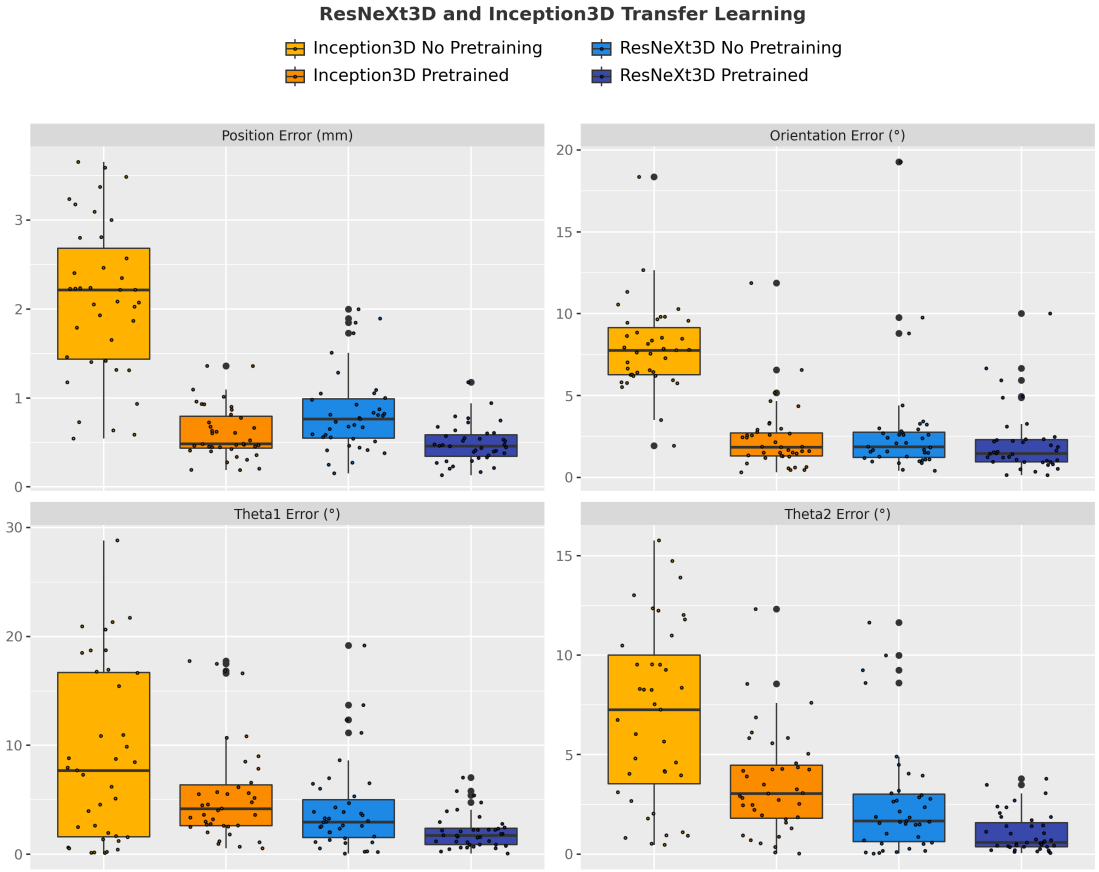

Figure 9. Performance comparison of Inception3D and ResNeXt3D with and without pretraining. Results highlight the considerable improvements achieved through pretraining.

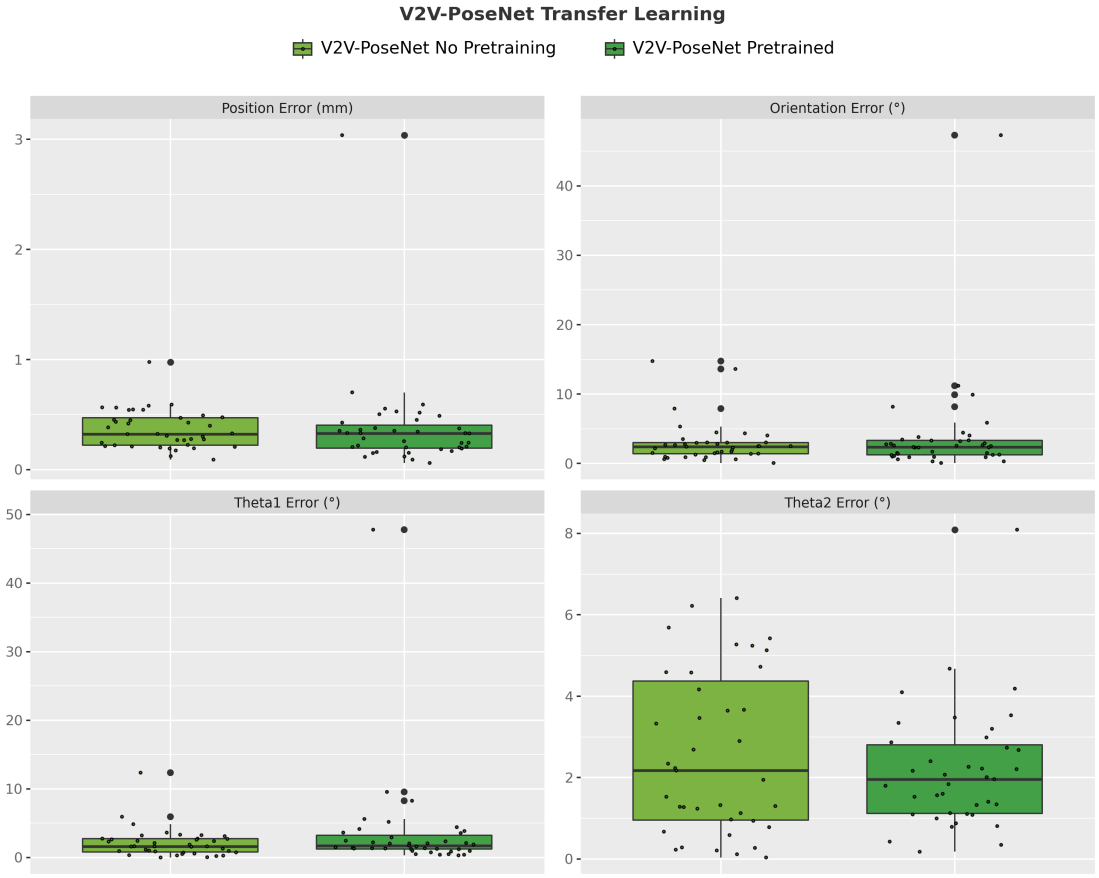

Figure 10. Performance comparison of V2V-PoseNet with and without pretraining. The results show no improvement from pretraining, indicating that V2V-PoseNet is less dependent on synthetic pretraining compared to Inception3D and ResNeXt3D.

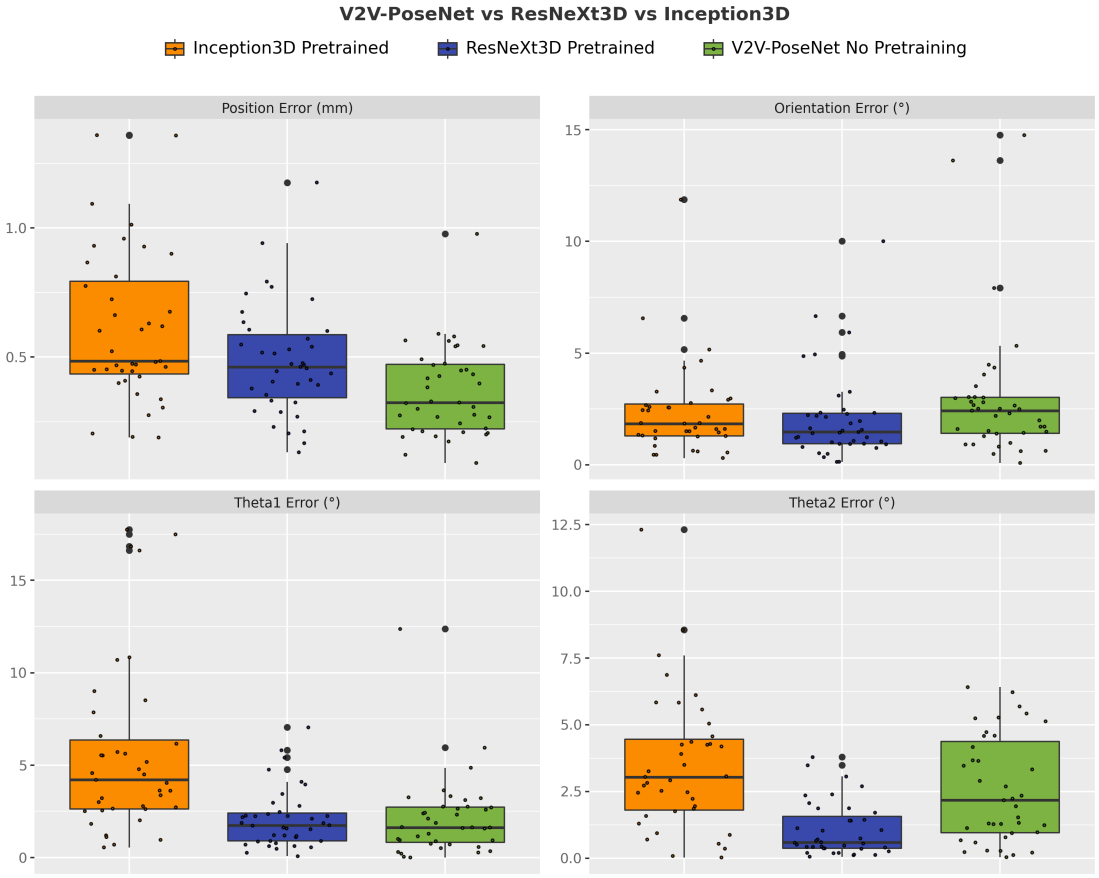

Figure 11. Performance comparison of V2V-PoseNet, ResNeXt3D, and Inception3D. V2V-PoseNet demonstrates superior position accuracy, while ResNeXt3D performs slightly better in orientation and joint angle estimation.

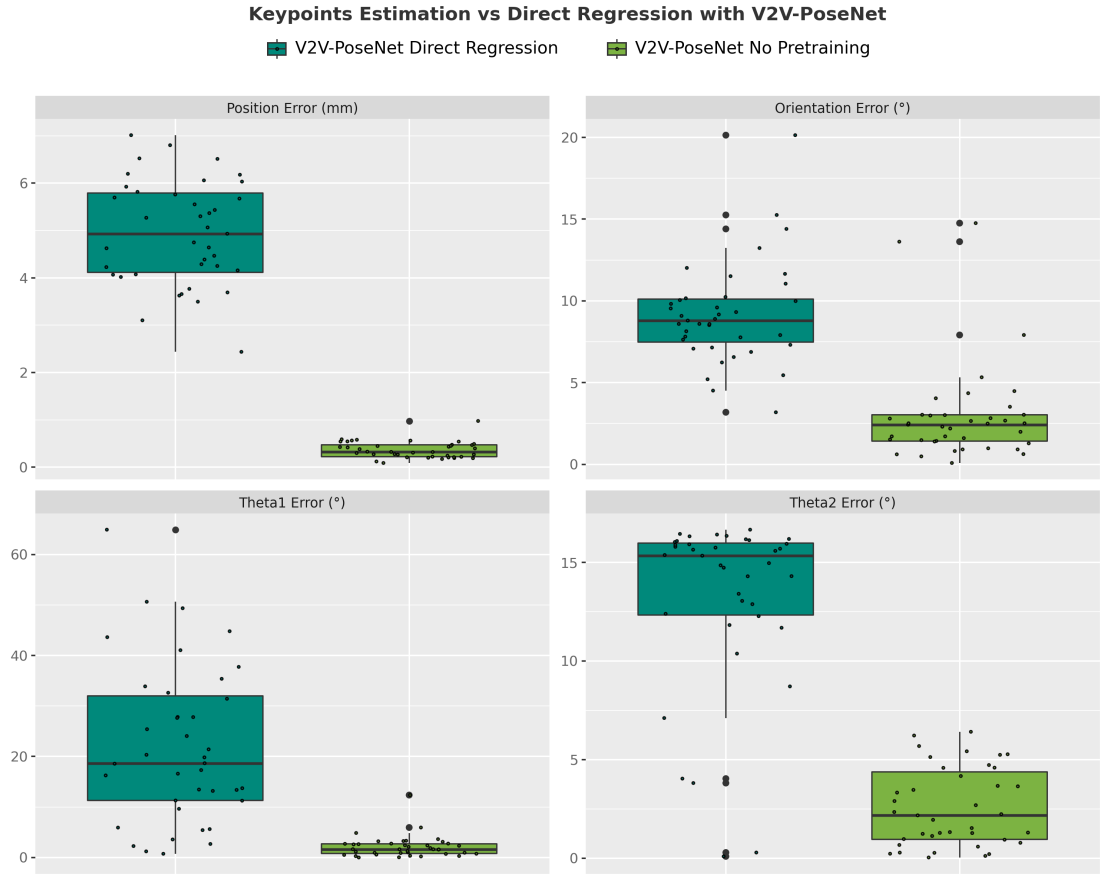

Figure 12. Comparison of V2V-PoseNet performance using direct regression versus keypoint estimation. The results highlight the considerable deterioration in performance when replacing keypoint estimation with direct regression.

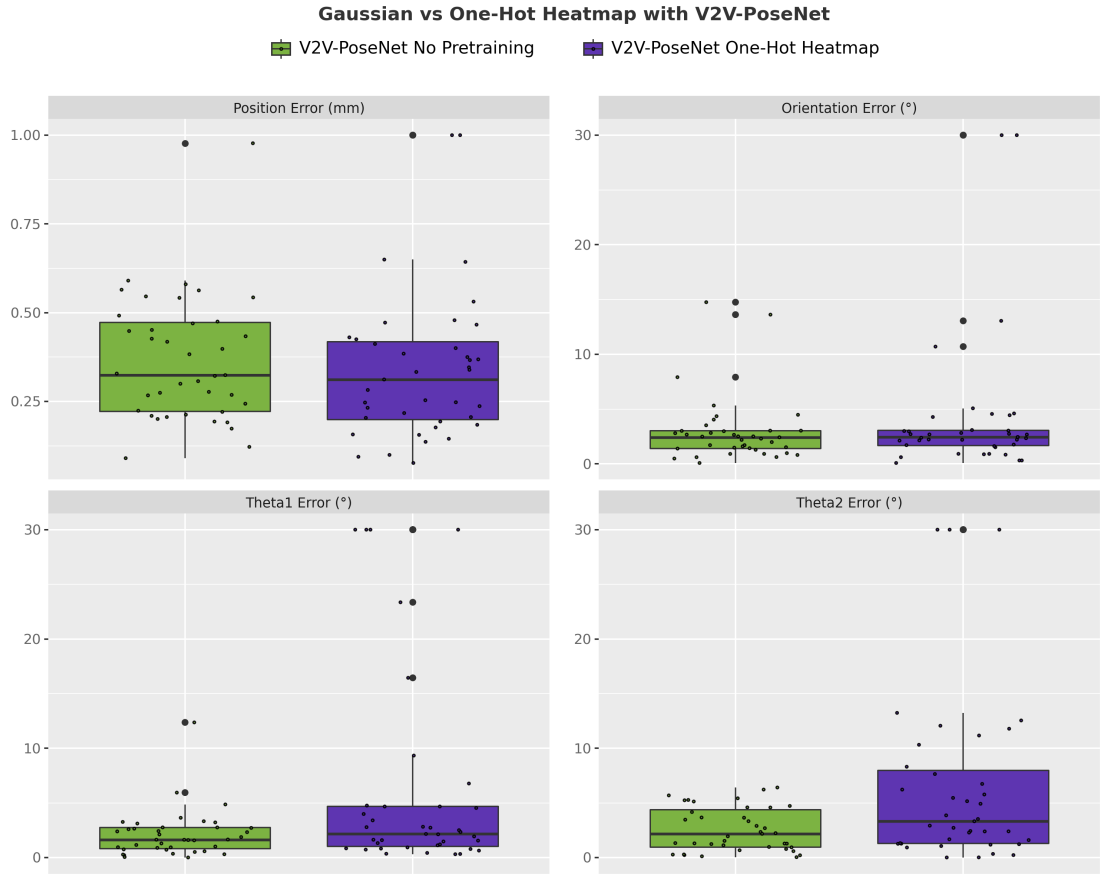

Figure 13. Comparison of V2V-PoseNet performance using Gaussian heatmaps versus one-hot heatmaps. Gaussian heatmaps outperform one-hot heatmaps, suggesting that the richer supervision provided by Gaussian heatmaps enables better spatial learning.

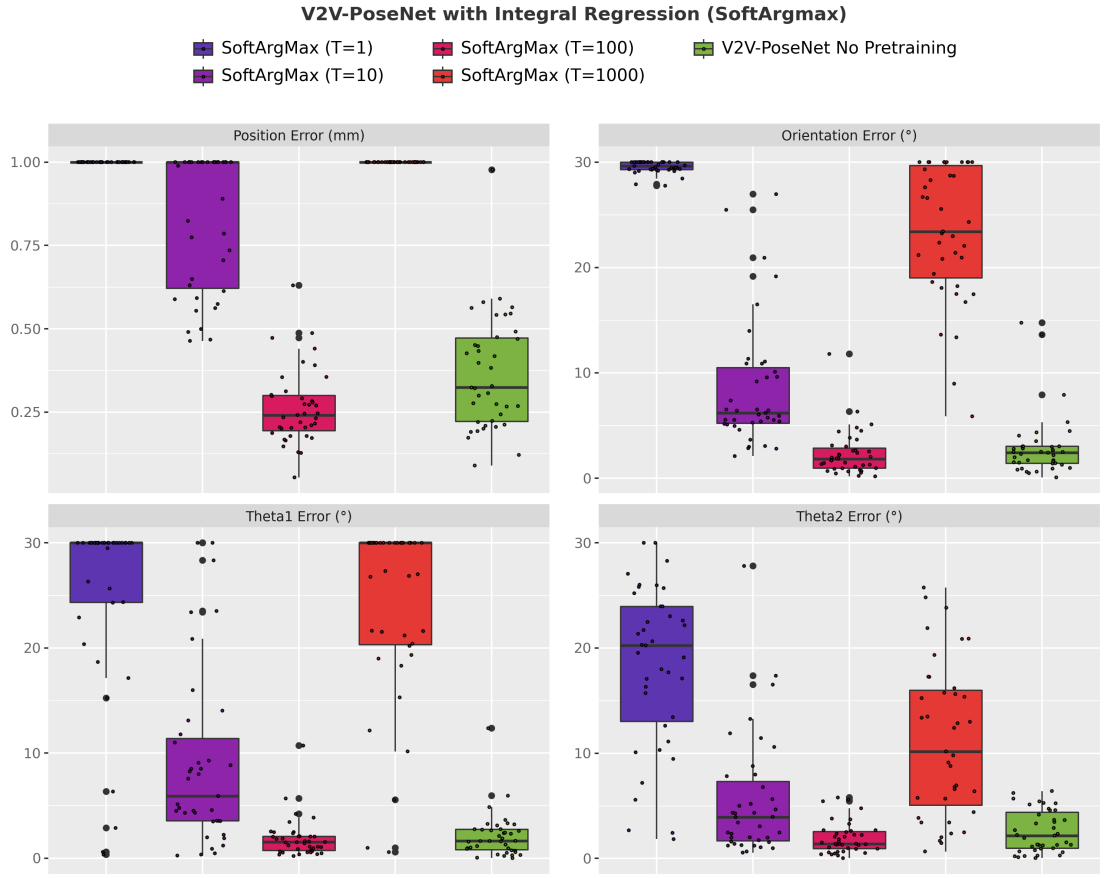

Figure 14. Performance comparison of V2V-PoseNet with SoftArgmax across different temperature values. A temperature of 100 achieves the best balance between gradient flow and smoothness of probability distribution in SoftMax, resulting in performance that surpasses the original V2V-PoseNet.

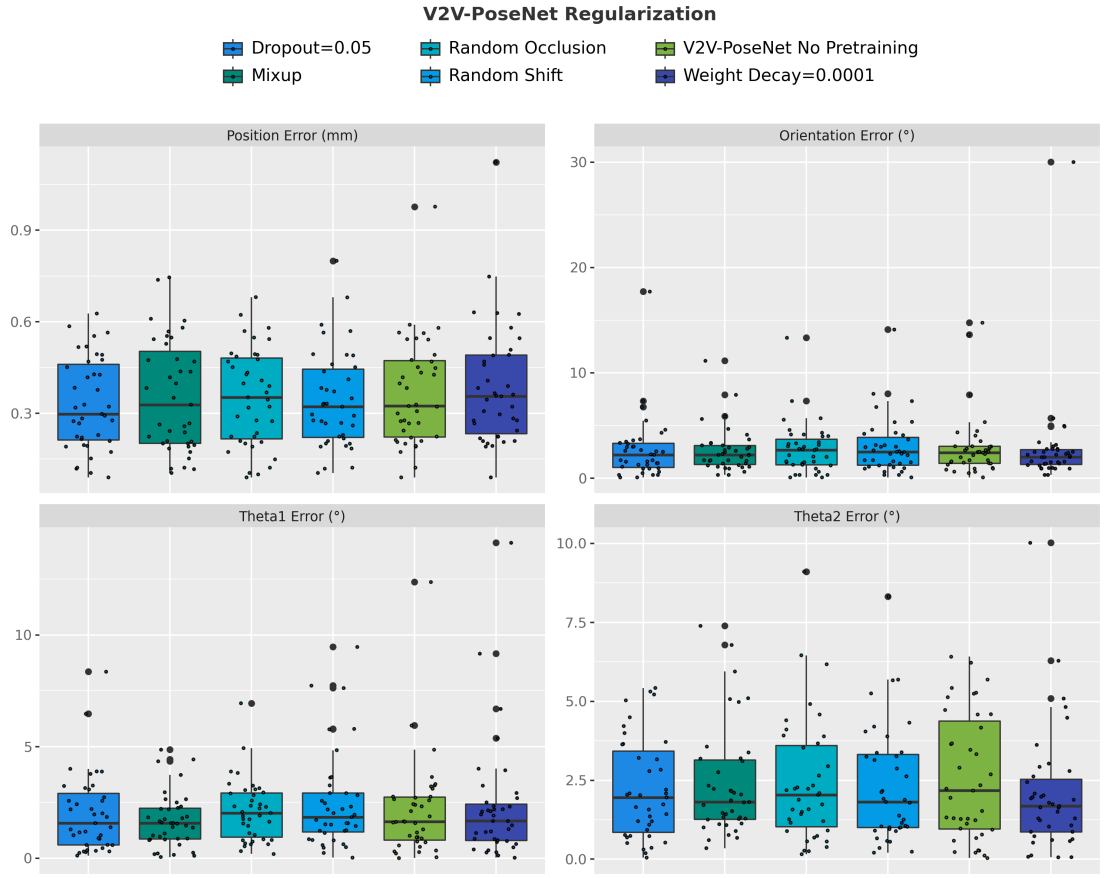

Figure 15. Comparison of V2V-PoseNet performance under various regularization techniques. The results indicate that none of the techniques yielded a considerable improvement over the baseline performance.

II. Licence

Non-exclusive licence to reproduce thesis and make thesis public

I, **Anton Zaliznyi**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to

   reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright,

   Real-time Pose Estimation of a Surgical Tool using Optical Coherence Tomography,

   supervised by Dr. Dmytro Fishman and Dr. Lueder Kahrs.

2. I grant the University of Tartu a permit to make the work specified in p. 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 3.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in p. 1 and 2.

4. I certify that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Anton Zaliznyi
30/12/2024