ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

Department of Industrial Engineering – DIN
Second Cycle Degree in Mechanical Engineering

Accurate Localization of 2D Ultrasound Probes Using Outside-In Tracking for Scanning the Neonatal Hip

Graduation Thesis in
Biomechanics

Supervisor	**Co-supervisor**	**Defended by**
Prof. Luca Cristofolini	Prof. Antony Hodgson, Prof. Marco Viceconti, Prof. Marco Palanca	Samuele Vaccari

Graduation Session: March 2026

Academic Year: 2025–2026

Fail again, fail better.

Contents

List of Figures

List of Tables

Acknowledgements

I would like to express my warmest thanks to my host supervisor, Professor Antony Hodgson, for guiding and supporting me throughout the development of this project. By giving me the freedom to work independently while always being available with sharp, timely insights when I felt lost, he helped me learn how to tackle a project of this scale.

I would also like to thank my home supervisor, Professor Luca Cristofolini, for his guidance and support throughout the different stages of writing this project.

A special thanks goes to Alix and Jakub for helping me understand the project and for their support during some particularly challenging moments.

I am grateful to my labmates: Liam, Emily, Ashvin, Cassidy, Henry, Dominik, Melissa, Winnie, Chantelle, and Rohan, for creating a welcoming, supportive, and stimulating lab environment.

My thanks also go to Luis for being a great mentor during my impostor syndrome phase and for generously letting me use the translational stage.

Thanks to Altea for reminding me when my work-life balance was slipping and for keeping me grounded along the way.

Finally, I am deeply grateful to my family and friends for their constant presence throughout this long, challenging, and rewarding journey, especially during the time I spent abroad.

Abstract

ABSTRACT BREVE

La displasia evolutiva dell'anca (DEA) è una frequente condizione neonatale in cui una diagnosi tardiva può determinare complicanze permanenti. Se non trattata, la DEA può causare claudicazione, discrepanza nella lunghezza degli arti, dolore, disabilità e osteoartrosi. Sono disponibili procedure di screening ecografico sia bidimensionali sia tridimensionali. L'ecografia 3D può ridurre la variabilità delle misurazioni rispetto all'imaging 2D, fornendo contesto volumetrico e piani di valutazione standardizzati. Tuttavia, i sistemi di ecografia 3D restano costosi e non sono ampiamente accessibili nella maggior parte dei contesti clinici.

Per colmare questo divario di accessibilità, il Surgical Technologies Lab ha sviluppato la Tracked 2D Ultrasound (TUS), che consente ai clinici di sintetizzare volumi 3D a partire da sonde ecografiche 2D standard mediante tracciamento ottico. Il laboratorio ha dimostrato che questo approccio raggiunge un'accuratezza sub-millimetrica e sub-grado nella stima della posizione, ma introduce alcune limitazioni pratiche nel flusso di lavoro clinico.

Per superare tali vincoli, abbiamo implementato un approccio di tracciamento Outside-In che utilizza due marker nel campo visivo della fotocamera: uno fissato alla sonda ecografica e uno al paziente. Questo assetto è pensato per migliorare l'ergonomia della sonda riducendone peso e baricentro, consentire un uso più versatile e permettere all'ecografista di concentrarsi sul paziente, venendo al contempo avvisato di eventuali movimenti del paziente durante la scansione tramite il secondo marker.

Il sistema di tracciamento impiega HydraMarkers, marker fiduciali flessibili che possono essere conformati alla geometria della sonda e restano rilevabili anche in presenza di occlusioni parziali, consentendo una stima della posa robusta pur adattandosi a superfici curve.

Il sistema è stato validato attraverso test sperimentali in condizioni clinicamente rappresentative. Le prestazioni sono state valutate lungo tutti e sei i gradi di libertà in termini di accuratezza traslazionale e rotazionale, ripetibilità e robustezza. L'approccio Outside-In ha raggiunto un'accuratezza traslazionale sub-millimetrica e un'accuratezza rotazionale sub-grado, supportando una ricostruzione 3D affidabile per la valutazione della DEA.

SHORT ABSTRACT

Developmental Dysplasia of the Hip (DDH) is a common neonatal condition in which delayed diagnosis can lead to lifelong morbidity. If untreated, DDH can cause limping, leg-length discrepancy, pain, disability, and osteoarthritis. Bi- and three-dimensional ultrasound screening procedures are available. 3D ultrasound can reduce measurement variability compared to 2D imaging by providing volumetric context and standardized assessment planes. Despite this, 3D ultrasound systems remain expensive and are not widely accessible in most clinical settings.

To address this accessibility gap, the Surgical Technologies Lab developed Tracked 2D Ultrasound (TUS), which enables clinicians to synthesize 3D volumes from standard 2D ultrasound probes using optical tracking. Although the Lab has demonstrated that this approach achieves sub-millimetric and sub-degree accuracy in position estimation, it introduces practical limitations in clinical workflow.

To overcome these constraints, we implemented an Outside-In tracking approach using two markers within the camera's field of view: one affixed to the ultrasound probe and another to the patient. This is expected to improve probe ergonomics by lowering its weight and lowering its center of mass, to enable more versatile probe use, and to allow the sonographer to concentrate on the patient while being notified of patient motion during the scan via the second marker.

The tracking system employs HydraMarkers, flexible fiducial markers that can be conformed to the probe geometry and remain detectable under partial occlusion, enabling robust pose estimation while accommodating curved surfaces.

The system was validated through experimental testing under clinically representative conditions. Performance was assessed across all six degrees of freedom in terms of translational and rotational accuracy, repeatability, and robustness. The Outside-In approach achieved sub-millimetric translational and sub-degree rotational accuracy, supporting reliable 3D reconstruction for DDH assessment.

EXTENDED ABSTRACT

Developmental Dysplasia of the Hip (DDH) affects up to 3% of the registered world cases. Early intervention through non invasive management, such as Pavlik harnesses, yields superior outcomes compared to surgical correction during adolescence. Therefore, accurate diagnosis and continuous monitoring throughout treatment are essential. If untreated, it can lead to limping, leg-length discrepancy, pain, and disability (El-Hariri et al. 2021). Over treatment risks over constraining the femoral head, potentially causing osteoarthritis at an earlier age, underscoring the importance of reliable screening and quantitative assessment.

Ultrasound imaging is the gold standard for neonatal hip assessment as it avoids radiation exposure and provides excellent visualization before ossification of the hip joint. Clinicians employ established assessment techniques, such as the Graf method based on the alpha angle measurement, to diagnose and classify DDH severity. While 2D ultrasound remains widely available and cost effective, it suffers from significant variability in extracted metrics due to interobserver differences, probe positioning, and lack of spatial context. Three dimensional ultrasound substantially reduces this variability by providing volumetric data and enabling standardized measurement planes. Nevertheless, 3D ultrasound systems are prohibitively expensive and deployed only at select regional centers, leaving most rural and community clinics without reliable access to standardized DDH assessment protocols.

The Surgical Technologies Lab addressed this accessibility challenge by developing Tracked 2D Ultrasound (TUS) technology. This approach works by mounting a small optical camera onto the 2D ultrasound probe, which tracks a reference marker placed in the examination environment. As the clinician moves the probe over the patient's hip, the camera continuously estimates the probe's pose relative to the marker, enabling offline synthesis of 3D volumetric reconstructions from sequential 2D ultrasound frames.

The Lab demonstrated that this Inside-Out system achieves sub-millimetric and sub-degree positional accuracy (Piwowarczyk et al. 2025). Accuracy was validated by simultaneous tracking of the ultrasound probe using both the proposed Inside-Out tracker and the NDI Vega across multiple motion trajectories: purely translational, purely rotational, and combined motion patterns representative of clinical hip scanning.

Despite strong technical performance, clinical implementation revealed significant practical limitations. The reference marker, a diamond shaped structure comprising multiple ArUco markers on a $30 \times 30 \times 30$ cm frame, occupies substantial space on the patient bed and physically obstructs the scanning area. Furthermore, the large structure constrains the sonographer's positioning freedom and inadvertently restricts probe approach angles. From an ergonomic standpoint, the mounted assembly also increases the effective weight and shifts the probe's center of mass away from the natural grip, making handling less balanced, more fatiguing, and less intuitive during routine scanning. Most critically, the Inside-Out approach cannot detect patient motion during scanning and if a parent's arm inadvertently moves into the line of sight of the camera, or parts of the marker move out of frame, tracking is lost mid-scan. These factors create workflow friction, particularly for experienced sonographers with established scanning protocols, making clinical adoption difficult.

A slightly different method that tackles some of these limitations has been developed, validated

and is being tested in vivo. This uses a ChArUco board which is a combination of Aruco markers and a chessboard. This tracking method allows partial occlusion of the board while still enabling estimation of the pose of the ultrasound probe in the world frame. It offers improved detectability due to the checkerboard and improved pose robustness due to the greater number of ArUco patterns. In addition, because it is a simple 30 × 20 cm board, it can be mounted on a tablet holder and clamped with a rig arm on the side of the bed, giving the sonographer more freedom and reducing the likelihood that parents are in the line of sight of the camera.

Nevertheless, two limitations remain, and they are intrinsic of the Inside-Out approach. First, because the camera is mounted on top of the probe, there may be confusion between probe movement and rigid motion of the whole setup, which could be patient motion. Second, 300 grams positioned 300 mm from the probe's transducer raise the center of mass of the instrument, affecting the natural scanning motion of the sonographer. For these reasons, and to improve versatility, the setup is switched to an Outside-In approach. This follows the same physics of the Inside-Out, but the camera is positioned on a rig, and two markers are in its field of view: one placed on the ultrasound probe and the other on the patient. This is expected to improve ergonomics by lowering weight and reducing the high center of mass, increase versatility in probe use, and allow the sonographer to concentrate on the patient while using the second marker to notify the user of patient motion during the scan.

To pursue this, HydraMarkers are used. These are a novel class of deformable, position-sensing markers with multifold local uniqueness properties. Unlike rigid ChArUco boards, which must be planar and printed on flat substrates, HydraMarkers can be fabricated in arbitrary shapes, bent, and attached directly around the ultrasound probe geometry. They remain robust under partial occlusion due to their inherent position sensing design, where every sub-region of the marker is unique in all four viewing orientations, enabling continued pose estimation even with incomplete marker visibility.

The system was validated through controlled experimental testing designed to replicate clinically relevant scanning scenarios. Performance assessment encompassed all six degrees of freedom, three translational axes and three rotational axes, to evaluate both accuracy and repeatability by comparing tracking results against stage based ground truth measurements. In addition, two distinct robustness assessments were performed. First, occlusion tolerance testing determined the minimum visible area of the HydraMarker required before the detection algorithm fails, quantifying the marker's resilience to partial visibility. Second, multi-marker discrimination testing verified the algorithm correctly distinguishes between the probe marker and patient marker when both are present in the field of view, ensuring unambiguous pose assignment.

The Outside-In approach with HydraMarkers demonstrated sub-millimetric translational accuracy and sub-degree rotational accuracy across all six degrees of freedom, meeting clinical requirements for DDH assessment.

These results establish the Outside-In architecture as a technically viable and clinically practical alternative to both expensive 3D ultrasound systems and the Inside-Out predecessor. By combining improved ergonomics, patient motion detection, and robust flexible markers, this approach removes previous workflow barriers and positions tracked 2D ultrasound as an accessible, high-quality diag-

nostic tool for DDH screening across a broader range of clinical settings. Future work will focus on refining the patient marker and integrating motion compensation strategies to account for involuntary movement during scanning. In parallel, efforts will aim to further enhance system robustness, enable real-time processing for immediate clinical feedback, and validate the complete framework in a broader in vivo clinical setting.

Chapter 1

Introduction

1.1 Problem and Clinical Need

1.1.1 Burden and Clinical Consequences of DDH

Developmental dysplasia of the hip (DDH) encompasses a spectrum of anatomical abnormalities ranging from mild acetabular dysplasia to complete hip dislocation and affects approximately 1% to 3% of newborns worldwide (Al Azri et al. 2023; Loder et al. 2011; Tao et al. 2023). When undetected or inadequately treated in early infancy, DDH can result in substantial long-term morbidity. Affected individuals frequently develop gait abnormalities, chronic pain, and early-onset hip osteoarthritis, with many ultimately requiring total hip arthroplasty during young adulthood (Bakarman et al. 2023; Schwend et al. 2014). The clinical and economic consequences of delayed diagnosis are severe, as late-detected DDH often necessitates surgical intervention at costs estimated to be approximately seven times higher than those associated with early conservative management (Harper et al. 2021; Woodacre et al. 2014). In contrast, diagnosis within the first three to six months of life enables conservative treatment with the Pavlik harness, which achieves successful hip reduction in 80% to 97% of cases when initiated during this optimal period (Atalar et al. 2007; Xu et al. 2020). Ultrasound screening performed after the fourth week of life provides sufficient diagnostic accuracy to guide timely intervention (Kelley et al. 2019).

The importance of early diagnosis is further underscored by the risks associated with both undertreatment and overtreatment. Inadequate or delayed management may lead to

residual acetabular dysplasia, progressive joint incongruity, and premature degenerative arthritis requiring surgical reconstruction (Bakarman et al. 2023; Schwend et al. 2014). Conversely, excessive or prolonged use of the Pavlik harness, particularly in hips that fail to achieve reduction, significantly increases the risk of avascular necrosis (AVN) of the femoral head (FH), an irreversible complication that compromises long-term hip function (Barney et al. 2025; Pap et al. 2006; Tiruveedhula et al. 2015). Studies report AVN incidence ranging from 0% with successful early Pavlik treatment to 28% following failed harness treatment or prolonged use beyond eight months (Pap et al. 2006; Tiruveedhula et al. 2015; Türközü et al. 2020). Failure to achieve concentric reduction within three weeks and treatment duration exceeding eight to nine weeks are established risk factors for AVN development (Pap et al. 2006; Tiruveedhula et al. 2015).

This narrow therapeutic window, in which early diagnosis enables effective conservative treatment while delayed or mismanaged intervention leads to severe complications, establishes a clear clinical need: DDH screening must be early, reliable, and accessible, relying on objective and reproducible diagnostic criteria to optimize patient outcomes and minimize the risks associated with inappropriate treatment.

1.1.2 Current Screening Pathways

Primary risk factors for DDH include breech presentation, positive family history, female sex, and primiparity, with incidence varying across populations (Bakarman et al. 2023; Loder et al. 2011). Breech presentation represents the most significant modifiable risk factor, conferring a three- to fourfold increased risk of DDH compared to vertex presentation, with risk magnitude increasing with the duration of breech positioning (Konijnendijk et al. 2023; Laskaratou et al. 2024; Tirta et al. 2025). Female infants exhibit an approximately fivefold increased risk, while a positive family history in first-degree relatives is associated with a twofold increase in incidence (Loder et al. 2011). Current screening pathways reflect this stratification, with high-risk newborns undergoing targeted assessment combining clinical examination and ultrasound imaging (Schwend et al. 2014).

Clinical screening relies on the Ortolani and Barlow maneuvers, provocative physical examination techniques intended to identify hip instability or dislocation in neonates. However, clinical examination demonstrates limited sensitivity, particularly when per-

formed by less experienced clinicians, and repeated forceful testing has been associated with iatrogenic hip instability (Sulaiman et al. 2011). Ultrasound has therefore become the imaging standard for DDH assessment, providing radiation-free visualization of the cartilaginous hip prior to femoral head ossification (Graf 2006). Screening is typically performed at four to six weeks of age, allowing physiologic hip immaturity present at birth to resolve while enabling reliable detection of structural abnormalities (Kelley et al. 2019; Laskaratou et al. 2024). Screening strategies vary internationally, ranging from universal ultrasound to selective imaging of high-risk infants combined with universal clinical examination, with effectiveness commonly evaluated using surgical intervention rates as a proxy outcome (Godward et al. 1998; Luoto et al. 2025; Roovers et al. 2005). Ultrasound-based screening demonstrates higher sensitivity than clinical examination alone (88.5% versus 76.4%), enables earlier detection, and reduces unnecessary referrals, though these benefits depend on access to specialized imaging infrastructure and trained personnel (Roovers et al. 2005).

1.1.3 Access Inequities and the Role of Imaging Infrastructure

A critical barrier to effective DDH screening lies in the geographic concentration of trained sonographers and pediatric radiologists at tertiary medical centers, creating substantial access inequities for rural and remote populations globally (Adams et al. 2021; Clarke-Grant 2025). Access inequities in DDH screening are not unique to any single healthcare system. A scoping review of DDH screening protocols across the 25 highest-ranked countries on the Bloomberg Global Health Index found that only four nations employ universal ultrasound screening, while five had no searchable screening programme at all, revealing persistent international inconsistency even among countries with the strongest health systems (Zusman et al. 2024). In Europe, a systematic review of 18 countries confirmed this fragmentation: six employ universal ultrasound, twelve use selective protocols, and the timing, technique, and referral criteria vary substantially between nations (Krysta et al. 2024). In low- and middle-income countries, the situation is markedly more severe: India currently lacks a national DDH screening programme, and a recent study found that children with DDH in India were diagnosed at a mean age of 24.7 months, well past walking age, with 76% ultimately requiring open surgery (Johari et al. 2023). A multi-organisational Delphi initiative, co-led by investigators at

the University of British Columbia, subsequently developed a context-specific DDH care pathway for India, acknowledging that care pathways designed in high-income settings lack utility in settings where 69% of the population is rural and ultrasound access is restricted by legislation and infrastructure constraints (Li et al. 2021).

In the Canadian context, Indigenous communities in northern and remote regions face compounding barriers to ultrasound access: geographic isolation necessitates air travel over distances exceeding 500 kilometers to reach imaging facilities, while financial constraints, fear of air travel, unfamiliarity with urban centers, and separation from family support systems create additional deterrents to seeking care (Adams et al. 2021; Clarke-Grant 2025; Morton et al. 2024).

Within British Columbia specifically, point-of-care ultrasound availability in rural communities remains limited: lack of specialized training and equipment cost represent the primary barriers cited by rural practitioners, with conventional ultrasound systems ranging from $25,000 to over $200,000 and advanced three-dimensional (3D) ultrasound platforms, which offer improved diagnostic reliability, exceeding $100,000 (Dr Sono 2025; Morton et al. 2024; Ott 2024). Training requirements compound these infrastructure challenges: achieving competency in infant hip ultrasound according to the Graf method requires a minimum of 50 supervised examinations for orthopedic surgeons and 200 supervised examinations for pediatric specialists, with certification to advanced proficiency standards (DEGUM Level III) demanding 400 documented ultrasound assessments (Bucher et al. 2021; Kraus et al. 2024). Families living in rural British Columbia report significant logistical and financial burdens associated with accessing DDH ultrasound screening at distant tertiary centers, including transportation costs, time away from work, childcare coordination challenges, and prolonged waiting times for appointments (Smith 2020). An estimated 1,800 infants annually in British Columbia require DDH ultrasound screening based on established risk criteria, yet geographic and infrastructure constraints delay or preclude timely assessment for a substantial proportion of this population [1] (Island Social Trends 2025; Smith 2020).

This clinical access problem creates an urgent need: DDH screening must become

[1]This estimate is derived by combining recent provincial birth statistics with published risk-factor prevalence. British Columbia has on the order of 38,000–41,000 births per year (Island Social Trends 2025). Breech presentation occurs in approximately 3% to 4% of term pregnancies (Gray et al. 2022), and additional high-risk categories together account for a further few percent of births (Kural et al. 2019; Loder et al. 2011). Assuming that 4% to 5% of newborns meet criteria for targeted ultrasound screening yields roughly 1,500-2,000 infants per year.

more accessible, reproducible, and deployable in non-tertiary settings without relying exclusively on expert sonographers or prohibitively expensive imaging platforms. These structural constraints are mirrored, in different forms, across other high-income countries, where DDH screening remains non-standardised (Krysta et al. 2024; Zusman et al. 2024),and are further compounded in low- and middle-income settings, where the absence of systematic screening results in diagnosis delays of years rather than weeks (Johari et al. 2023), underscoring the global urgency for screening solutions that are robust to variability in imaging infrastructure and operator expertise. Addressing this need requires technological and methodological innovations that maintain diagnostic accuracy while reducing both operator dependency and infrastructure complexity.

1.2 Previous Approaches

1.2.1 2D Ultrasound and the Graf Method

Two-dimensional ultrasound (2DUS) is the current standard imaging modality for diagnosing DDH in infants, providing real-time visualization of the cartilaginous acetabulum and proximal femur without ionizing radiation (Graf 2006; Starr et al. 2014). In the widely adopted Graf method, hips are assessed using a standardized coronal plane through the deepest point of the acetabulum, with the ilium aligned vertically to form a reproducible anatomical reference (Graf 2006; Radiologyassistant.nl 2018). Within this plane, three key lines are defined: the baseline along the lateral ilium, the bony roof line along the osseous acetabular roof, and the cartilaginous roof line through the labrum (Graf 2006). Their intersection yields the alpha and beta angles, which quantify bony and cartilaginous coverage of the femoral head, respectively, and form the basis of the Graf classification into normal, immature, and dysplastic hip types (Graf 2006; Starr et al. 2014). Femoral head coverage, measured as the percentage of the femoral head contained within the acetabulum, is frequently used as a complementary metric and correlates positively with the alpha angle (Gunay et al. 2009). Thresholds of approximately 51% and 39% femoral head coverage have been reported to correspond to mature and pathological hip development, respectively (Gunay et al. 2009). Figure 1.1 summarizes the plane and angle definitions.

Despite its widespread adoption, the reliability of 2DUS in DDH assessment is con-

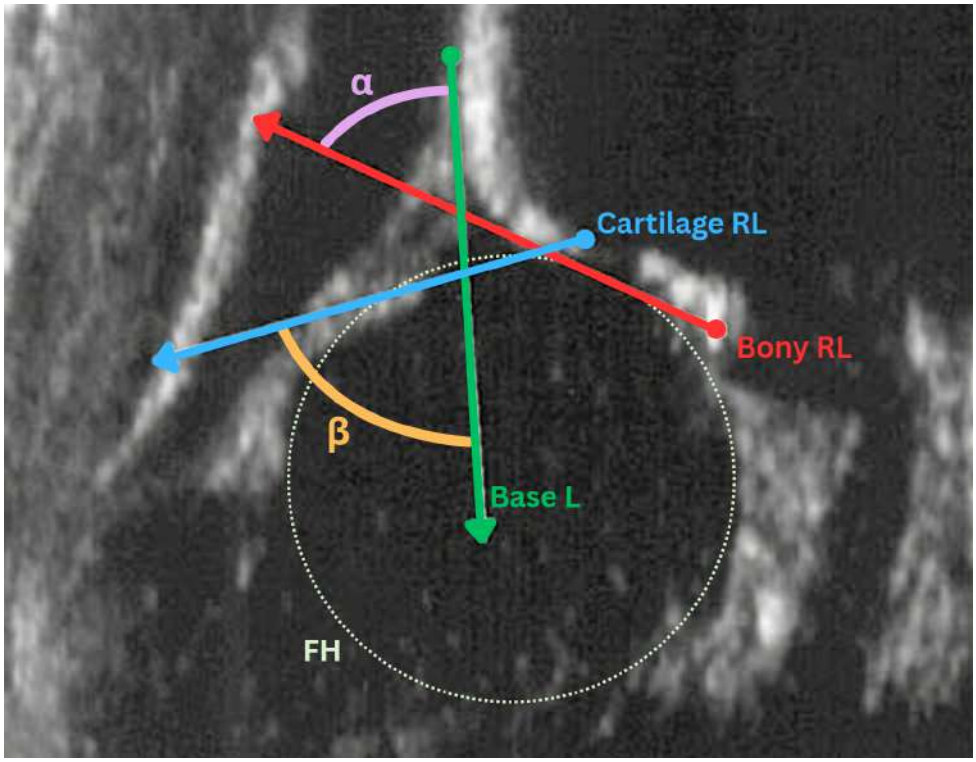


Figure 1.1: Graf plane and angle definitions. The baseline (green), bony roof line (red), and cartilaginous roof line (blue) define the alpha (pink) and beta (orange) angles, measured relative to the baseline. The femoral head (FH) is overlaid with a light green outline. Adapted from Graf (2006).

Table 1.1: Synopsis of sonographic hip types (Graf 2006).

Category	α angle	β angle	Interpretation
Normal (Type I)	$\alpha \geq 60°$		Mature acetabulum; adequate osseous coverage.
Immature/borderline (Type II)*	$50° \leq \alpha \leq 59°$		Reduced osseous roof development; follow-up depends on age and clinical context.
Dysplastic (Type IIc/D)	$43° \leq \alpha \leq 49°$	IIc: $\beta < 77°$ D: $\beta > 77°$	Marked bony deficiency; increased risk of instability/decentration.
Dislocated (Type III–IV)	$\alpha < 43°$		Severe dysplasia with displacement patterns; requires specialist management.

*Type II is commonly interpreted in relation to age (physiological immaturity in early weeks versus persistent immaturity beyond this period) (Graf 2006).

strained by its dependence on strict plane selection and operator expertise. Obtaining a true Graf standard plane is technically demanding: small deviations in probe orientation can alter the apparent inclination of the acetabular roof and labrum, violating quality criteria and changing the measured alpha angle (Graf 2006). Three-dimensional ultrasound resection studies have demonstrated that acceptable 2D images meeting Graf criteria can be acquired over a range of probe tilts averaging $24°$, within which the measured alpha angle can vary by up to $19°$ for the same hip purely due to changes in probe orientation (Jaremko et al. 2014). This variability is sufficient to shift a hip between normal, immature, and dysplastic categories in more than half of examined cases, leading either to overdiagnosis with unnecessary treatment or to underdiagnosis with missed pathology (Jaremko et al. 2014). Comparative studies between novice and expert operators underscore this operator dependency: novices performing 2DUS hip screening produce significantly lower image quality scores (2.7 versus 4.9 on a 5-point scale), demonstrate poor inter-rater reliability (ICC 0.35), and generate misdiagnoses in up to 39% of examined hips, whereas the same novices achieve near-expert reliability when using 3D ultrasound with automated plane selection (Mostofi et al. 2019). Furthermore, retrospective analyses of hips initially classified as Graf Type I that subsequently deteriorated reveal that these cases were attributable to technical errors in plane acquisition and failure to adhere to Graf quality checklists rather than true progression of normal hips, illustrating the consequences of inadequate operator training (Gunay et al. 2019). Consequently, while 2DUS and the Graf method are relatively low cost and globally accessible, their diagnostic performance is limited by operator-dependent plane selection and angle measurement, especially outside high-volume specialist centers (Bucher et al. 2021; Starr et al. 2014).

1.2.2 3D Ultrasound: Performance Improvements vs. Accessibility Limitations

Three-dimensional ultrasound (3DUS) represents a volumetric imaging modality that captures the complete hip anatomy in a single sweep. Mechanically swept probes perform rapid sequential scanning of adjacent planes, while newer matrix array transducers capture volumetric data in real-time, both enabling post-acquisition computation of anatomical metrics from any arbitrary plane within the captured volume (Tsedendamba

et al. 2025). This fundamental architectural difference directly reduces operator dependency: instead of acquiring a single 2D image that must satisfy strict plane criteria, 3DUS operators capture a volume and software subsequently identifies the optimal imaging plane, effectively decoupling diagnostic accuracy from the expertise required during image acquisition (Quader et al. 2021).

Evidence from comparative studies demonstrates substantial improvements in measurement reliability and diagnostic accuracy when 3DUS is applied to DDH screening, particularly for novice operators. In direct comparisons between 2DUS and 3DUS, inter-rater reliability for alpha-angle measurement improved from poor (ICC 0.10–0.35) with 2DUS to moderate-to-high (ICC 0.73–0.84) with 3DUS among novice operators (Geng et al. 2020; Mostofi et al. 2019). Novice operators using 3DUS correctly classified 57 of 58 (98%) infant hips, with zero misclassifications, compared to 8 Type I and 33 Type II misclassifications (39% error rate) when performing 2DUS (Mostofi et al. 2019). These improvements are attributed to automated plane selection within the volumetric dataset, which substantially reduces measurement variability associated with probe orientation (Quader et al. 2021).

Despite these diagnostic advantages, 3DUS remains inaccessible as a population-level screening solution. Three-dimensional ultrasound systems require significantly higher capital investment than conventional 2DUS equipment, with advanced volumetric platforms costing substantially more than the $100,000–$200,000 range of standard ultrasound systems (Dr Sono 2025; Ott 2024). The physical footprint, cooling requirements, and technical complexity of 3DUS platforms further limit deployment to specialized imaging centers. Consequently, 3DUS expertise and equipment remain heavily concentrated at tertiary medical centers, reinforcing the geographic access disparities described in Section 1.1.3. These limitations have prompted investigation of alternative approaches, including machine learning and artificial intelligence systems designed to enhance diagnostic reliability while maintaining equipment simplicity and accessibility (M. Chen et al. 2024; Jaremko et al. 2023).

1.2.3 AI-Assisted Diagnosis and Persistent Quality Dependency

Artificial intelligence and machine learning methods have been proposed to address operator dependency in DDH screening by automating landmark detection, angle measurement,

and classification. Deep learning architectures, particularly convolutional neural networks, have been trained on annotated ultrasound datasets to identify key anatomical structures and subsequently compute Graf alpha and beta angles, enabling automated assessment of hip morphology (Jaremko et al. 2023). Under controlled conditions and with high-quality input images, these systems demonstrate diagnostic performance approaching that of expert radiologists, with pooled sensitivity and specificity exceeding 90% in systematic reviews and meta-analyses of retrospective datasets (M. Chen et al. 2024). In feasibility studies conducted in primary care settings, AI-augmented portable ultrasound enabled nurses and family physicians with minimal ultrasound training to achieve DDH detection rates comparable to those of formal ultrasound screening performed by expert sonographers, suggesting that AI decision support could democratize access to reliable DDH screening (Jaremko et al. 2023). Ongoing clinical trials are evaluating AI-enhanced portable ultrasound systems in low-resource and community settings to determine whether these promising laboratory results translate into effective population-level screening tools (ISRCTN Registry 2024).

However, AI diagnostic performance degrades substantially when applied to suboptimal ultrasound acquisitions, revealing a fundamental limitation: AI systems trained to interpret anatomy cannot compensate for images in which that anatomy is inadequately visualized. Systematic evaluation of an AI diagnostic system (MEDO-Hip) demonstrated that measurement accuracy dropped from 89% on high-quality scans (quality score greater than 7 out of 10) to 57% on lower-quality scans (quality score ≤ 7), with 13 of 107 images deemed entirely uninterpretable by the AI due to motion artifacts and poor visualization of the os ischium (Hareendranathan et al. 2022). Specific failure modes include patient motion during acquisition, probe misalignment resulting in off-plane imaging, and acoustic shadowing that obscures critical landmarks, all of which occur more frequently in non-specialist clinical environments where infants may be uncooperative and operators lack extensive experience (Hareendranathan et al. 2022). AI-based quality assessment systems have been developed to provide real-time feedback, alerting operators when acquired images fall below diagnostic thresholds and prompting rescanning (Hareendranathan et al. 2021).

Consequently, while AI assists in interpretation and can partially mitigate inter-observer variability in measurement, it cannot overcome the fundamental limitations of

poor image acquisition. The diagnostic accuracy of AI-assisted DDH screening remains contingent on the quality of the input ultrasound data, which in turn depends on operator skill in probe manipulation, patient cooperation, and adherence to standardized imaging protocols.

1.2.4 Inside-Out Tracked 2D Ultrasound and its Limitations

Tracked two-dimensional ultrasound offers a strategy to reconstruct three-dimensional volumes from conventional 2D probes by tagging each acquired frame with the probe's spatial pose, enabling volumetric assessment without requiring dedicated 3D ultrasound hardware. Tracked ultrasound systems adopt various localization technologies, including electromagnetic, optical (both Outside-In and Inside-Out configurations), and inertial sensors, to record probe position and orientation during scanning (Peng et al. 2022). Inside-Out optical tracking was introduced in the Surgical Technologies Laboratory as a promising approach for freehand ultrasound scanning, implemented with a camera mounted rigidly to the ultrasound probe that observes external fiducial markers positioned within the clinical environment (Piwowarczyk et al. 2025). Initial implementations employed a non-planar arrangement of three ArUco markers mounted in a cuboidal configuration (Fig. 1.2) to improve pose estimation robustness and reduce ambiguity. Laboratory validation of this configuration demonstrated sub-millimetric and sub-degree accuracy, establishing technical feasibility for neonatal hip imaging (Piwowarczyk et al. 2025).

The Inside-Out architecture offers practical advantages in portability, as the camera-probe assembly remains self-contained and does not require external cameras to view the entire clinical workspace. However, clinical deployment of the system for DDH screening has revealed substantial limitations that degrade reconstruction quality and compromise the reliability of extracted diagnostic metrics.

Phantom-based validation of Inside-Out tracking inherently excludes patient motion, yet neonatal hip scanning involves crying, squirming infants, involuntary movements, breathing artifacts, and postural shifts. These factors introduce pose uncertainties not captured in rigid phantom experiments and that would compound tracking errors during clinical deployment (Piwowarczyk et al. 2025). These anticipated challenges were confirmed in a clinical evaluation of an optically tracked ultrasound system (OTUS) at

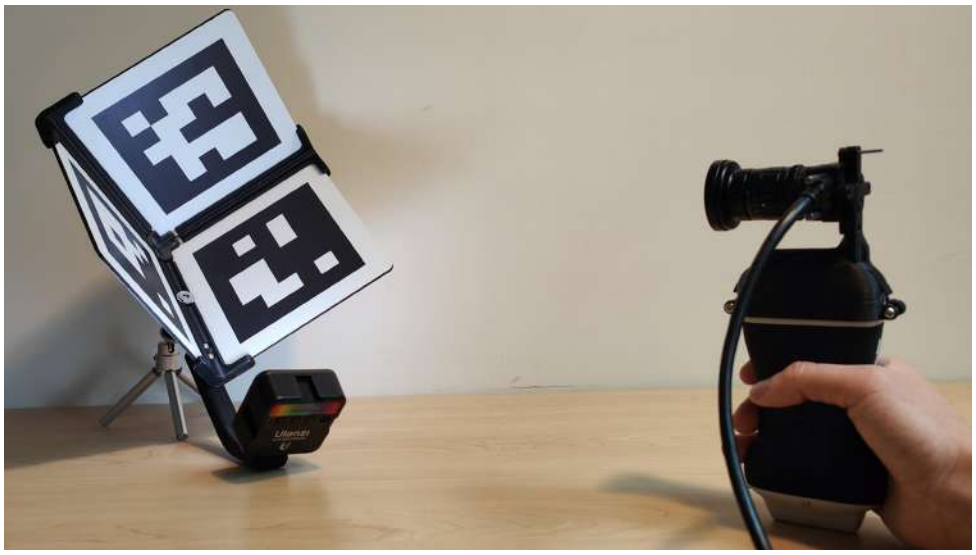


Figure 1.2: Inside-Out tracking setup with a Basler Dart camera, mounted on the Clarius ultrasound probe (right), oriented toward an illuminated Cuboidal 3D ArUco marker array (left) comprising three `DICT_5X5` ArUco markers mounted on a 3D-printed frame.

BC Children's Hospital, where Inside-Out tracking was deployed for DDH screening in routine pediatric patients (K. K. Chen 2025). Of 102 attempted scan acquisitions across 51 hips, 30 (29%) were excluded from analysis due to tracking failures or inadequate ultrasound content: 10 scans exhibited unusable tracking trajectories characterized by marker occlusions or implausible pose estimates, while 20 scans captured insufficient anatomical information (cropped ilium, inadequate femoral head visualization) to support diagnostic metric extraction (K. K. Chen 2025). Among the remaining 72 scans deemed processable, qualitative assessment revealed multiple categories of tracking degradation attributable to clinical environment factors. Line-of-sight occlusions, caused by parents reaching toward infants, clinician repositioning, or medical equipment movement, produced transient or sustained tracking loss that manifested as discontinuities or implausible jumps in reconstructed probe trajectories. Marker placement on compliant surfaces, such as the examination bed, introduced subtle marker motion that was not visually apparent to operators but caused jitter and reconstruction errors detectable in postprocessing. Real-time feedback on tracking quality was not integrated into the clinical workflow, leaving operators unaware of acquisition failures until postprocessing review revealed unusable data (K. K. Chen 2025).

1.3 Global Research Objective

The fundamental limitation of Inside-Out tracking is that it does not track the patient. Therefore, any patient motion during scanning is not accounted for and may introduce a potentially very large error in the estimated probe trajectory and reconstructed volume. In contrast, an Outside-In approach employs an external camera to simultaneously track markers attached to both the ultrasound probe and the patient anatomy. Rather than treating patient motion as noise to be minimized, this dual-marker approach would enable explicit measurement of patient movement, allowing real-time alerts to the sonographer and supporting future work on motion-compensated reconstruction algorithms.

The global research objective of this thesis is to investigate whether an Outside-In optical tracking architecture can provide accurate and robust localization of a 2D ultrasound probe when scanning the neonatal hip, establishing a technical foundation for subsequent motion-aware imaging.

1.4 Key Knowledge Gap

Outside-In optical tracking is a well-established approach in computer vision and image-guided surgery, in which fixed cameras observe fiducial markers attached to moving instruments. Conventional implementations rely on rigid planar markers such as ArUco tags, which have demonstrated sub-millimeter pose estimation accuracy under controlled laboratory conditions (Oščádal et al. 2020). However, the requirement for flat mounting surfaces restricts marker placement on curved ultrasound probe housings, while reliance on full marker visibility renders standard configurations vulnerable to partial occlusions common in clinical environments. Detection reliability further degrades at oblique viewing angles or when markers occupy a limited number of pixels in the image plane (Poroykov et al. 2020).

HydraMarkers have recently been proposed as a deformable, position-sensing fiducial field designed to address these limitations and are already available as an open-source toolkit for marker field generation (Zhu et al. 2023). Unlike rigid planar markers, HydraMarkers can conform to non-planar surfaces and maintain robust pose estimation under partial occlusion due to their intrinsic multifold local uniqueness properties. Prior work

has demonstrated that HydraMarker-based tracking achieves fractional-millimeter translational accuracy (RMS error $\leq 0.5\,\mathrm{mm}$) and sub-degree rotational accuracy at working distances of approximately $1.5\,\mathrm{m}$ using a stereo camera configuration (Xie et al. 2023). These results establish HydraMarkers as a promising candidate for applications requiring occlusion resilience and flexible marker geometry. Nevertheless, existing evaluations have been limited to stereo camera systems operating at relatively large working distances, leaving open the question of performance under substantially different imaging conditions.

In comparison, neonatal hip ultrasound scanning with an external camera positioned near the examination area typically operates at short working distances ($300\,\mathrm{mm}$ to $400\,\mathrm{mm}$), where a single-camera configuration becomes practically viable. At these distances, markers occupy a larger fraction of the image plane, potentially enabling accurate pose estimation without stereo triangulation. However, it remains unknown whether a single-camera, Outside-In HydraMarker-based tracking system can achieve sub-millimeter translational and sub-degree rotational accuracy sufficient for tracked 2D ultrasound reconstruction under these conditions. Moreover, the robustness of such a system to clinically realistic occlusion patterns, such as partial hand coverage and the wide range of probe orientations encountered during freehand neonatal scanning, has not yet been established.

1.5 Research Question

This thesis addresses the following research question, which directly targets the identified knowledge gap:

How well do HydraMarker-based Outside-In tracking systems perform in localizing a 2D ultrasound probe for neonatal hip scanning, in terms of rotational and translational accuracy, repeatability, and robustness to partial occlusion at clinically relevant working distances ($300\,\mathrm{mm}$ *to* $400\,\mathrm{mm}$*)?*

To address this overarching research question, this thesis investigates three specific sub-questions:

1. What rotational and translational accuracy can be achieved by a single-camera, Outside-In HydraMarker-based tracking system relative to ground-truth reference

measurements?

2. What is the repeatability of pose estimation across repeated trajectories?

3. How robust is the system to partial marker occlusion, and how does identification performance degrade when multiple markers are simultaneously present in the camera field of view?

The remainder of this thesis is organized as follows. **Chapter 2** presents the technical background of optical tracking systems, reviewing Outside-In and Inside-Out architectures, clinical-grade tracking technologies, and the evolution of vision-based fiducial markers from early binary patterns to modern position-sensing marker fields, culminating in the HydraMarker framework employed in this work. **Chapter 3** describes the system design and development, including hardware configuration, the dual HydraMarker strategy for simultaneous probe and patient tracking, marker design optimization balancing identification capacity and detection robustness, and the complete detection and identification pipeline. **Chapter 4** outlines the experimental setup, evaluation protocol, and evaluation metrics. **Chapter 5** reports validation results across all six degrees of freedom, including jitter analysis, rotational and translational accuracy, and robustness under partial occlusion and multi-marker conditions. **Chapter 6** discusses opportunities for improving detection robustness, achieving real-time performance, and outlines the clinical validation pathway required for translation to routine DDH screening. Finally, **Chapter 7** contextualizes the findings relative to existing tracking systems, evaluates the advantages of the proposed Outside-In architecture, identifies current limitations, and synthesizes the contributions of this work toward enabling motion-aware tracked ultrasound for accessible DDH screening.

Chapter 2

Background

2.1 Optical Tracking Systems

Optical tracking systems are fundamental technologies in image guided surgery, motion capture, and spatial localization applications. The fundamental principle underlying optical tracking is stereo triangulation. A calibrated stereo camera system, consisting of two or more cameras with precisely known intrinsic and extrinsic parameters, observes fiducial markers in a three dimensional space. The intrinsic parameters characterize each camera's internal geometry (focal length, principal point, distortion coefficients), while extrinsic parameters define the spatial relationship (rotation and translation) between the reference frames of the cameras.

While optical tracking systems may employ two or more cameras, the triangulation principle is most easily illustrated for a stereo pair. In the following, we therefore consider two cameras and denote the corresponding image points as $\mathbf{p}_1$ and $\mathbf{p}_2$. For a point P in 3D space, each camera projects this point onto its 2D image plane according to the perspective projection model. Given the 2D projections $\mathbf{p}_1$ and $\mathbf{p}_2$ observed in camera 1 and camera 2 respectively, and the calibration matrices of both cameras, the 3D position of P can be recovered through triangulation. This is formalized via epipolar geometry: for a point observed in one camera, epipolar constraints define a one-dimensional locus (the epipolar line) in the other camera's image where the corresponding point must lie. In the stereo case, the intersection of the two corresponding epipolar lines yields the 3D position (Hata et al. n.d.; Scaramuzza n.d.), thus enabling three dimensional localization and motion tracking in global coordinates.

Camera calibration is prerequisite to triangulation and is typically performed using planar calibration targets (e.g., checkerboards) imaged from multiple viewpoints, from which intrinsic and extrinsic parameters are estimated via algorithms, such as Zhang's method (Open CV Documentation 2024a).

2.1.1 Outside-In Tracking

Outside-In tracking is an architecture in which fixed cameras observe moving fiducial markers attached to instruments or anatomical landmarks, as exemplified by optical tracking systems such as Polaris Vega (retroreflective spheres), Optotrak Certus (active LED markers), and MicronTracker (planar fiducials). As the instrument moves, the cameras continuously observe and localize the markers via triangulation (for multiple cameras) or via PnP problem-solving (if sufficient markers are visible to a single camera). The computed marker positions are then used to infer the instrument's pose in the fixed world coordinate frame.

This architecture offers significant advantages. The system exhibits robust occlusion tolerance: if markers are distributed on the instrument, partial occlusion still permits accurate pose estimation. Moreover, the system can detect independent environment motion; since the camera frame of reference is fixed, any movement of the environment or patient is distinctly identified. Instrument design is simplified, as these require only passive markers without active electronics or cameras. The use of camera triangulation and multi-marker redundancy generally enables high accuracy. Additionally, the system offers excellent scalability: multiple instruments can be tracked simultaneously by a single camera pair using distinct marker sets.

Despite these advantages, considerable infrastructure is required. External cameras must be carefully positioned and calibrated relative to the workspace, and the system is inherently dependent on fixed camera placement. Any camera repositioning necessitates complete recalibration. Consequently, the tracking volume is constrained by the configuration of the cameras (their positions, orientations, and overlapping fields of view), and setup complexity is significant, as precise positioning is required prior to use, with any subsequent misalignment requiring recalibration. In addition, costs are higher due to the need for multiple cameras, rigging hardware, and calibration equipment.

Such systems dominate several application domains. In industrial applications, robotic

arms are often calibrated using camera based tracking systems that observe tool markers. In biomechanics, gait analysis and sports motion analysis rely on Outside-In optical tracking with retroreflective markers. In image-guided surgery, optical tracking systems are widely used to localize surgical instruments and anatomical landmarks in real time, enabling accurate navigation by correlating the physical scene with preoperative or intraoperative imaging data.

2.1.1.1 Clinical-Grade Outside-In Optical Trackers

The Polaris Vega series, manufactured by Northern Digital Inc. (NDI), represents the gold standard for clinical grade optical tracking in image-guided surgery. These systems are widely integrated as the tracking backbone of commercial image-guided surgery and robotic navigation platforms, and dedicated Polaris Vega variants (such as the Vega XT and Vega ST) are explicitly positioned for surgical applications rather than purely experimental use (NDI 2022; NDI 2024).The system employs two high resolution infrared cameras operating at up to 400 Hz (Vega XT) that detect passive retroreflective spherical markers affixed to surgical instruments or anatomical landmarks.

The Vega XT achieves a positional accuracy of $0.12\,\mathrm{mm}$ RMS for marker localization and a rotational accuracy of approximately 0.05° RMS, with marker configurations spanning $50\,\mathrm{mm}$ to $100\,\mathrm{mm}$. The positional repeatability is $0.10\,\mathrm{mm}$ RMS with rotational repeatability of 0.05° RMS. The system supports a working volume of up to $2.5\,\mathrm{m} \times 2.5\,\mathrm{m} \times 2.5\,\mathrm{m}$ and operates with markers typically $4.8\,\mathrm{mm}$ to $9.5\,\mathrm{mm}$ in diameter (NDI 2022). These retroreflective coated spherical beads reflect incident infrared light back toward the integrated light source, creating bright spots in both camera images. High resolution imaging sensors detect these circular projections and compute their centroids to sub pixel precision. Stereo triangulation then combines the 2D centroids from both cameras to recover the 3D position of each marker. When multiple markers are rigidly affixed to an instrument (as illustrated in Fig. 2.1), their relative 3D positions encode the instrument's orientation and the six degree of freedom pose.

The Polaris Vega's clinical success stems from several key advantages, starting from the sub-millimeter positional and sub-degree rotational accuracy, which exceed most clinical requirements. The retroreflective property ensures robust, high-contrast marker detection and rapid centroid computation through spherical symmetry. Finally, the passive design

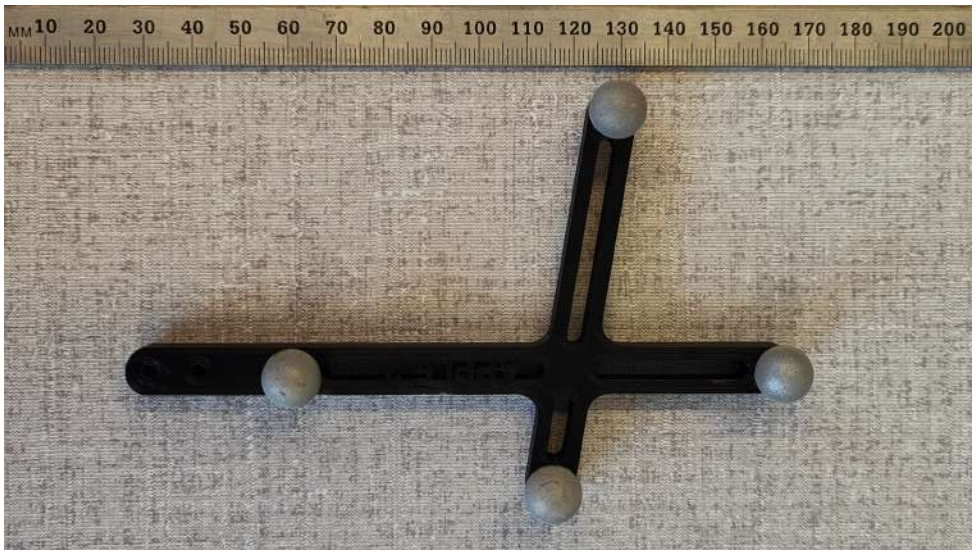


Figure 2.1: Passive tool with retroreflective spherical markers.

simplifies instrument design and sterilization since it doesn't require active electronics or batteries.

The fixed spherical geometry constrains marker design, as adapting marker configuration or size requires mechanical redesign of instrument mounts. The tracking system is fragile to occlusion; loss of any single marker degrades pose estimation and requires sufficient marker redundancy (typically 3–4 markers per instrument) for robustness. Markers must be rigidly mounted to instruments, restricting placement flexibility in confined surgical spaces. Marker density is also limited since triangulation requirements and occlusion considerations prevent arbitrary close placement. Finally, commercial retroreflective markers and surgical instrument mounts incur significant costs, making the system expensive for multi-instrument surgical workflows.

The Vega ST variant is a compact, portable version designed for surgical applications with similar accuracy specifications but reduced working volume (NDI 2024), shown in Fig. 2.2.

Technical validation studies confirm these specifications. A peer reviewed technical assessment of the Polaris Vega system reported positional accuracies consistently below $0.2\,\mathrm{mm}$ and rotational accuracies below 0.3° across typical surgical tracking scenarios (Fattori et al. 2021).

Beyond NDI, other commercial optical tracking systems exist for medical applications. The Optotrak Certus (NDI) uses active infrared emitting markers (LEDs). This approach provides higher accuracy ($\approx 0.10\,\mathrm{mm}$ positional accuracy) and operates at higher frame

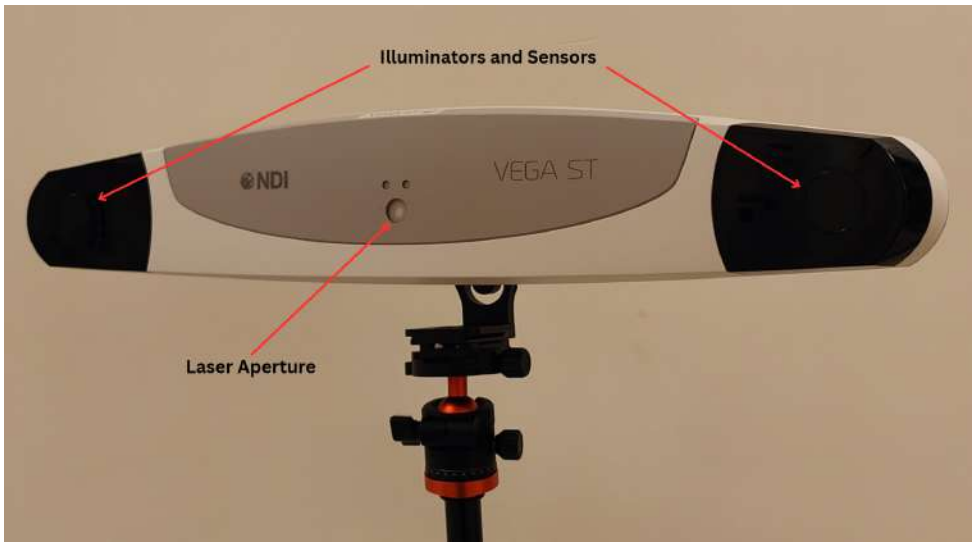


Figure 2.2: Front view of the NDI Vega ST optical tracking camera, highlighting the illuminators and sensors on both ends and the central laser aperture.

rates (up to $4500\,\mathrm{Hz}$) but requires powered markers, which can be disadvantageous for certain applications (NDI n.d.; Schmidt et al. 2009). The MicronTracker (Claron Technology) is a camera-based optical tracker that uses planar fiducial markers (similar to AprilTags or ArUco). This system achieves sub-millimeter accuracy and is particularly suited for applications where rigid marker attachment is not feasible (Technology 2023).

2.1.2 Inside-Out Tracking

Inside-Out tracking employs a camera affixed to an instrument that observes stationary fiducial markers (ArUco, AprilTag, and ChArUco boards) placed in the environment. By tracking the visual position of these markers across successive camera frames, the system computes the relative pose between the camera, and thus the instrument, and the environment. This computational problem is equivalent to solving the Perspective-n-Point problem: given n correspondences between 3D world marker positions and 2D image coordinates, the camera's six degrees of freedom pose is estimated.

The primary benefit of this system is portability, as only the instrument requires augmentation (camera + mount), while no external infrastructure is needed. The working volume is flexible and is only limited by the marker visibility range, which can be large. Additionally, calibration is simplified as there is no need to calibrate the relative poses of multiple external cameras. Furthermore, Inside-Out systems can be implemented with low cost components such as a standard webcam and printed ArUco markers, making

them economically accessible.

Nevertheless, this architecture lacks the ability to decouple instrument motion from external motion of the tracked system. Additionally, integrating a camera onto the instrument increases its mass and shifts its center of gravity, potentially degrading ergonomics, precision during manual surgical tasks, and instrument mobility in a clinical setting.

Inside-Out tracking is widely employed across diverse domains. In augmented reality, mobile phones and AR headsets (e.g., HoloLens) use on-device cameras to localize the device relative to environmental features. In robotics and navigation, mobile robots employ visual odometry to navigate indoor environments using visual landmarks. In virtual reality, VR controllers often integrate this approach to estimate hand pose relative to markers fixed in the room.

Despite their architectural differences, all optical tracking systems share fundamental limitations inherent to the imaging modality. A universal constraint is the line of sight requirement between the tracking camera(s) and fiducial markers. Consequently, occlusion by surgical instruments, hands, drapes, or anatomical structures disrupts tracking regardless of system architecture. Passive optical systems are additionally susceptible to environmental factors such as sensitivity to reflective surfaces and ambient infrared sources. All optical systems require precise camera positioning and careful calibration before surgery, and any bumping or misalignment of cameras necessitates re-calibration.

2.2 Electromagnetic Tracking Systems

Electromagnetic (EM) tracking systems provide an alternative to optical tracking by generating a quasi-static or dynamic magnetic field of known geometry using a transmitter. Receiver coils, positioned on the instruments to be tracked, experience an induced voltage proportional to the magnetic field at their location and orientation. Through calibration, the six degrees of freedom pose of each coil relative to the transmitter can be inferred from the measured magnetic field (NDI 2023a; NDI 2023b), thereby determining the pose of the instrument.

The Aurora system, NDI's EM tracking solution for surgical guidance, provides positional accuracy between $0.5\,\mathrm{mm}$ and $1.5\,\mathrm{mm}$, depending on the working volume and sensor orientation, and rotational accuracy of 1° to 2°. The working volume is approxi-

mately $1.0\,\mathrm{m} \times 1.0\,\mathrm{m} \times 1.0\,\mathrm{m}$, slightly smaller than typical optical systems such as Polaris Vega. Update rates can reach up to $240\,\mathrm{Hz}$, and miniature receiver coils, ranging from 2 to 5 mm, can be embedded within surgical instruments or catheters, minimizing instrument perturbation and enabling minimally invasive applications (NDI 2019; NDI 2023a). Unlike optical systems, Aurora's magnetic field penetrates soft tissue and non-conductive materials, making it insensitive to occlusion by surgical drapes, instruments, or personnel. The trade-off is substantially lower positional accuracy than optical tracking systems and sensitivity to ferromagnetic materials. In fact, common operating room equipment such as metal instruments and monitors can distort the magnetic field and degrade performance. Additionally, the inherent non-uniformity of the magnetic field requires frequent recalibration with changes in environmental setup or instrument positioning, and imposes constraints on the proximity and composition of surrounding equipment, since ferromagnetic objects and powered imaging devices (e.g. C-arms, operating tables, surgical instruments) can distort the field and degrade tracking accuracy.

A peer reviewed comparative study of optical and electromagnetic tracking systems in surgical applications found that optical systems achieved $0.1\,\mathrm{mm}$ to $0.2\,\mathrm{mm}$ positional accuracy while EM systems achieved $0.5\,\mathrm{mm}$ to $1.0\,\mathrm{mm}$ accuracy. EM systems showed superior performance only in scenarios with significant line of sight occlusion, such as endoscopic surgery with heavily occluded instruments (Koivukangas et al. 2013).

2.3 Vision-Based Fiducial Markers

Whereas NDI uses passive retroreflective or active LED markers, other applications rely on fiducials which are detectable and trackable by standard cameras. Fiducial markers are artificial visual landmarks designed for automatic detection, identification, and localization in digital images. Unlike natural features that depend on environmental texture and illumination, fiducial markers provide controlled, distinctive patterns that enable rapid and robust feature detection with minimal computational overhead. Their development has been driven by two primary constraints: limited camera field of view and severe visibility problems such as occlusion, making them essential in applications ranging from camera calibration and augmented reality to robot navigation and surgical guidance (Fan et al. 2018; Hutchinson et al. 1996; Kim et al. 2015; Watanabe et al. 2017; Zhang 2000).

2.3.1 Early Marker Systems: Repeated Entities and Landmark-Based Approaches

In controlled environments where markers are fully visible, such as calibration tasks, the simplest approach involves repeated geometric entities like crosses, squares, dots, or concentric circles. These require no encoding because features can be identified by their order or spatial relationships. However, as applications demanded operation in less controlled conditions with partial occlusion, the limitations became apparent: without encoded information on individual features, nearby features cannot be reliably identified when a portion of the marker is hidden. This motivated the introduction of coded patterns attached to repeated entities, creating a two-tier system where landmarks or geometric relationships guide identification to nearby coded features, as in early ARToolKit markers (Fig. 2.3 (1)) (Kato et al. 1999). For instance, coded squares, circles, or dot patterns were placed alongside reference structures, allowing the system to first locate the reference, then identify coded information in its vicinity. While this approach improved robustness compared to entirely unmarked markers, it still failed when the field of view became very small or occlusions severe enough to obscure both references and nearby features simultaneously.

2.3.2 ARTag as Self-Identifying Binary Markers

The transition to fully self-identifying markers represented a fundamental shift. Rather than depending on context or proximity to landmarks, each captured region must encode its own identity. One of the earliest and most influential approaches was ARTag (Fig. 2.3 (2)), which adopted a square planar design with a black outer border and an interior 6×6 binary grid (36 bits) encoding the marker identity (Fiala 2005). This system introduced forward error correction, allowing markers to be reliably identified even when some bits were corrupted (Fiala 2010). However, it did not guarantee robust separation between all tags under rotation, creating potential for inter-tag confusion as a corrupted bit pattern from one tag could be incorrectly decoded as another tag.

The detection algorithm for binary square markers follows a consistent pipeline. First, the image gradient is computed at every pixel. A graph-based clustering algorithm groups pixels with similar gradient directions and magnitudes, forming edge segments. These

segments are then assembled into candidate quadrilaterals by searching for sequences of four segments that form closed four sided shapes. The quad's corners, derived from line intersections, provide sub-pixel localization. A homography transformation is computed from the quad's corners and the known square layout, allowing mapping between marker coordinates and image coordinates. Using this homography, the interior bit pattern is sampled and thresholded to binary values. Finally, the bit pattern is compared against a codebook of valid identifiers to determine if the detection is valid and to identify which marker was detected (Fiala 2005).

2.3.3 AprilTag: Lexicode-Based Generation and Improved Detection

A significant advancement came with AprilTag (Fig. 2.3 (3)), which introduced lexicode-based tag generation with explicit Hamming distance guarantees (Olson 2011). Lexicodes are generated by iteratively adding candidate codewords to a codebook, ensuring that each new codeword maintains at least a specified minimum Hamming distance from all previously added codewords. Crucially, AprilTag extended this to guarantee that all four rotations of a codeword maintain the same minimum Hamming distance from all other tags' rotations. Additionally, this architecture incorporated a geometric complexity heuristic during codebook generation: candidate tags with very simple patterns (e.g., solid black or simple stripes) are rejected because such patterns occur frequently in natural imagery and lead to false positives. The complexity is measured as the minimum number of rectangles needed to describe the tag's 2D pattern; a tag with higher complexity is less likely to arise accidentally in natural scenes.

For AprilTag's 36-bit coding scheme with minimum Hamming distance 10 (denoted 36h10), theoretical false positive rates are extremely low: $1.0 \times 10^{-6}\,\%$ with 0 bit errors corrected, rising to $4.95 \times 10^{-1}\,\%$ with 5 bit errors corrected (Olson 2011). In practical experiments on the LabelMe dataset of 1.81×10^{5} natural images (containing no real AprilTags), the empirical false positive rate with up to 2 bit errors corrected was measured at $5.70 \times 10^{-4}\,\%$, matching the theoretical prediction (Olson 2011). When compared to alternative coding schemes, AprilTag 36h10 achieved 2221 unique codes with minimum Hamming distance 10, outperforming ARTag (2046 codes, distance 4), and ARToolkitPlus variants (512–4096 codes with distances 2–4) (Fiala 2005; Olson 2011).

The detection algorithm for AprilTag similarly employs gradient-based edge detection

and quad-finding, but with important refinements (Olson 2011). The image is convolved with a Gaussian filter to reduce noise while preserving large scale edge information. Pixels are clustered into components with similar gradient direction and magnitude using a graph-based agglomerative clustering method. Line segments are fit to each component using weighted least squares, with weights corresponding to gradient magnitude. Four sided shapes are then identified through a two steps search. This generates many false positive quad candidates, the subsequent decoding step filters out invalid detections. Additionally, to handle variations in illumination within a single marker, AprilTag uses a spatially varying threshold model learned from the marker's border pixels; a linear model $I(x, y) = Ax + Bxy + Cy + D$ predicts expected intensity values for both black and white pixels at each spatial location, with the threshold set at their midpoint (Olson 2011).

Localization accuracy is critical for pose estimation applications. In ray-traced synthetic experiments with known ground truth, AprilTag achieved significantly better localization accuracy than ARToolkitPlus across a range of viewing angles and distances (Olson 2011). When the marker normal was rotated up to 60° away from the camera, AprilTag maintained robust detection; ARToolkitPlus detection rate dropped sharply at similar angles. At fixed orientation, AprilTag reliably detected markers at distances of $50\,\mathrm{m}$ or more maintaining a low position error, whereas ARToolkitPlus dropped to below 50% detection rates around $25\,\mathrm{m}$ (Olson 2011).

AprilTag 2 and AprilTag 3 further refined the detection pipeline to improve speed and robustness (Wang et al. 2016). A key innovation was the continuous boundary segmentation algorithm, which groups edge pixels not by simple connectivity but by the pair of connected components they separate. This technique is more resistant to noise and to the merging of boundaries that are only a single pixel apart. The quad fitting step was reformulated to identify candidate corner points by analyzing mean squared error as a window of boundary points is swept around the marker's perimeter. Instead of exhaustively fitting lines to all possible four-point combinations, strong peaks in the MSE function are identified as corner candidates, and only permutations of these corners are tested. This reduces computational complexity while improving robustness (Wang et al. 2016).

Adaptive thresholding was employed, dividing the image into 4×4 pixel tiles and computing local minimum/maximum intensity to suppress low contrast regions from

further processing. These refinements reduced false positive rates further: on the LabelMe dataset, AprilTag 2 generated approximately 75% fewer candidate quads than the original AprilTag (13.6 million vs. 51 million candidate quads from 4.21×10^5 images), yet detected fewer false positives (6 vs. 145), demonstrating that the improved detector's selectivity reduced both the computational load and false positive rate simultaneously. Importantly, AprilTag 2 demonstrated that comparable localization accuracy could be achieved on decimated (half-resolution) images, enabling dramatic speedups (Wang et al. 2016).

2.3.4 ArUco Library: MILP-Based Dictionary Generation

Unlike AprilTag, which employs lexicode-based tag generation, the ArUco library adopts Mixed Integer Linear Programming (MILP) as its dictionary generation approach (Garrido-Jurado et al. 2014). MILP optimization minimizes the probability of inter-marker confusion by formulating and solving constraints that enforce specified Hamming distances between all marker pairs in a dictionary (Garrido-Jurado et al. 2015). This approach allows flexible customization of marker dictionaries based on application-specific requirements: the number of unique markers needed, the bit length of each marker (typically 4×4, 5×5, or 6×6 bits), and the desired inter-marker Hamming distance for error correction capability (Garrido-Jurado et al. 2014; Garrido-Jurado et al. 2015).

The ArUco library provides predefined dictionaries with various configurations, one of which is shown in Fig. 2.3 (4). Common examples include `DICT_4X4_50` (4×4 bits, 50 markers, minimum Hamming distance of 4), `DICT_5X5_250` (5×5 bits, 250 markers, minimum Hamming distance of 6), and `DICT_6X6_250` (6×6 bits, 250 markers, minimum Hamming distance of 4) (Garrido-Jurado et al. 2014). Additionally, larger dictionaries are available, such as `DICT_ARUCO_ORIGINAL` with 1024 markers of 5×5 bits and a minimum Hamming distance of 3. A critical design principle is the tradeoff between dictionary size and inter-marker distance: smaller dictionaries with fewer markers achieve higher Hamming distances and thus greater robustness to bit errors, while larger dictionaries support more markers but with reduced error correction capability (Garrido-Jurado et al. 2014; Garrido-Jurado et al. 2015).

The detection algorithm for ArUco markers follows a consistent pipeline similar to other binary square markers. The image is converted to grayscale, and a sequence of morphological operations and contour detection identifies candidate marker regions.

For each candidate, a quadrilateral is fit to the contour, providing corner localization. A homography transformation maps the marker's interior bit pattern to a normalized coordinate space. The binary pattern is thresholded and compared against the dictionary codebook to identify the marker ID and validate the detection (Garrido-Jurado et al. 2014). Despite these advantages, conventional planar ArUco markers remain sensitive to occlusion: even modest corner occlusion can lead to incorrect corner localization or complete detection failure, which directly degrades pose estimation (Adámek et al. 2023; OpenCV Development Team n.d.).

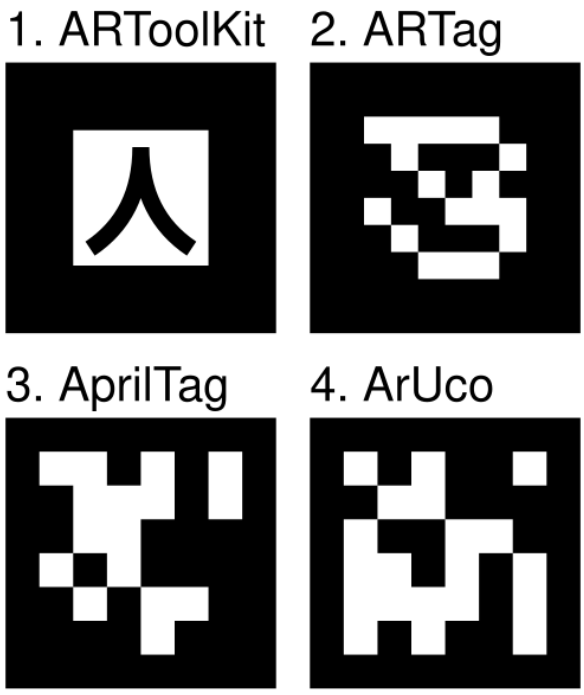


Figure 2.3: Comparison of augmented reality fiducial markers. Image by Cmglee, licensed under CC BY-SA 4.0.

2.3.5 ChArUco Boards: Combining ArUco and Chessboard Patterns

While standard ArUco markers provide reliable detection with flexible dictionary generation, their corner localization accuracy is limited by the discrete pixel-level resolution of the marker grid. ChArUco boards address this limitation by combining the flexibility of ArUco markers with the high-precision corner localization of traditional chessboard patterns (Garrido-Jurado et al. 2014). A ChArUco board is a hybrid target consisting of a chessboard grid where a subset of the white squares are replaced with ArUco markers of varying IDs, creating a unified pattern where both the ArUco markers and chessboard

structure are mutually supportive (see Fig. 2.4).

The key advantage of ChArUco boards lies in their superior corner accuracy. In standard chessboard patterns, every corner has two adjacent black squares, enabling precise sub-pixel refinement; however, chessboard patterns lack the flexibility of ArUco markers when partial visibility or occlusion is present, requiring the entire board to remain visible for reliable detection. Conversely, ArUco markers alone allow partial occlusion due to their independent encoding, but their corner positions have limited sub-pixel accuracy due to the marker's discrete internal grid structure. ChArUco boards exploit the strengths of both: the ArUco portion is used to interpolate the position of the chessboard corners, providing the versatility to tolerate occlusions or partial views, while the interpolated corners belong to a chessboard pattern, achieving sub-pixel accuracy far superior to standard ArUco corner detection (Garrido-Jurado et al. 2014).

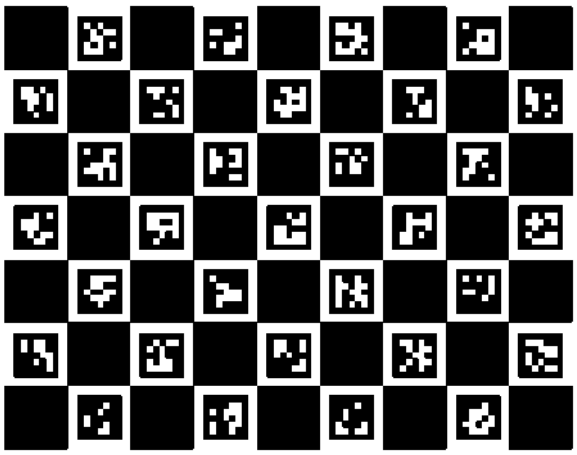


Figure 2.4: 7×9 ChArUco board using ArUco markers from the DICT_5X5_50 dictionary.

Detection of ChArUco boards proceeds in two stages. First, ArUco markers within the board are detected and localized using the standard ArUco pipeline. Next, the detected marker positions are used to interpolate the chessboard corner positions. If calibration parameters are available, the interpolation proceeds by estimating a rough camera pose from the detected ArUco markers, then reprojecting the ChArUco corner coordinates back to the image plane and refining them via sub-pixel optimization using local gradient information. If calibration parameters are not available, the interpolation calculates the homography between the ChArUco plane and its image projection, using only the ArUco markers closest to each chessboard corner to minimize the effects of image distortion (Garrido-Jurado et al. 2014).

The corner density of ChArUco boards is significantly higher than that of ArUco-only patterns. A single ArUco marker provides only four corner observations, whereas a ChArUco board with, for example, a 7×9 chessboard grid populated with ArUco markers (see Fig. 2.4) provides 80 corner observations per board detection. This high corner density, combined with sub-pixel interpolation, yields extremely accurate pose estimation and camera calibration results.

2.3.6 RuneTag: Cyclic Codes and Occlusion Resilience

RuneTag approached marker design from a coding theory perspective, applying cyclic error-correcting codes. Rather than a grid of squares, RuneTag consists of circular dots spatially arranged in concentric layers. The reciprocal positions of dots between layers are carefully chosen based on cyclic code properties, allowing the marker to be identified even with severe occlusions (Bergamasco et al. 2011).

Cyclic codes provide mathematical guarantees on error correction capability (Hamming 1950; Wicker et al. 2003). RuneTag's RUNE-129 design can identify approximately 2.0×10^4 distinct markers while sustaining up to 70% occlusion resilience (see Fig. 2.5) (Bergamasco et al. 2011). This exceptional occlusion robustness arises because cyclic codes allow any cyclic shift of a codeword to also be a valid codeword, and the mutual rotations and redundancy in the dot pattern enable partial decoding and error correction even with significant missing information.

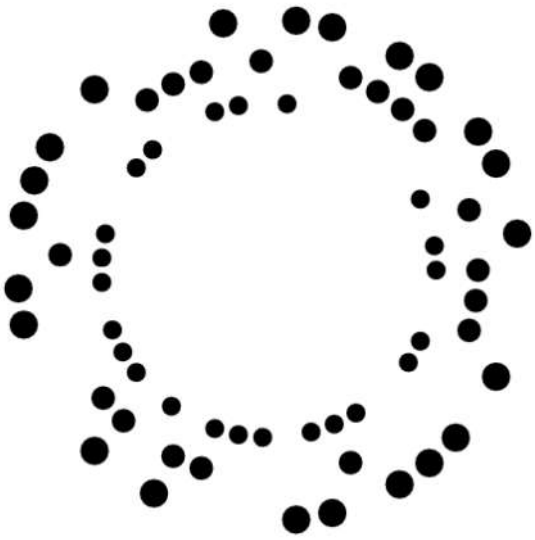


Figure 2.5: RUNE-129 fiducial marker example. Image by Bergamasco et al. (2011)

Detection of RuneTag proceeds by identifying the circular dot features in the image, computing their relative positions, and then matching these position patterns against the codebook of valid cyclic codes. The circular dot pattern provides stable feature localization across scale and viewing angle, contributing to RuneTag's documented robustness against noise, blur, and various distortion types (Bergamasco et al. 2011).

2.3.7 Circular Fiducial Markers: CCTag and High Accuracy Detection

While square binary markers dominated the field, circular markers offered distinct advantages for certain applications. CCTag is a fiducial system based on concentric circles with different crown ratios, designed for extremely high localization accuracy even under challenging imaging conditions (Calvet et al. 2016). The marker consists of alternating black and white concentric circles; the unique ratio of crown sizes encodes the marker identity, as shown in Fig. 2.6. Detection proceeds by identifying circle edges in the image and fitting ellipses to them, then measuring the radii to determine the crown pattern (Fitzgibbon et al. 2000).

Figure 2.6: CCTag marker examples with three crowns. Each marker is uniquely identified by the crown thickness, encoding a marker ID. Image from the CCTag documentation, used with permission (MPL 2.0) (Calvet et al. 2016).

In synthetic experiments with severe motion blur, CCTag successfully detected and identified markers with center localization accuracy of 0.54 pixels and 0.36 pixels under different blur conditions (Calvet et al. 2016). This level of accuracy under motion blur, a condition that severely challenges square markers, demonstrates the geometric stability of circular patterns. The circular design provides naturally symmetric gradient information, making center localization stable even under substantial deformation.

2.3.8 Topological Fiducials: reacTIVision and Graph-Based Recognition

An alternative paradigm for marker recognition emerged from topological approaches, exemplified by reacTIVision (Bencina et al. 2005a; Bencina et al. 2005b). Rather than encoding information in geometric patterns or binary codes, reacTIVision recognizes fiducials by their topological structure, specifically the containment relationships between black and white regions when the image is binarized, as illustrated in Fig. 2.7.

The detection algorithm first binarizes the image using adaptive thresholding. The thresholded image is then segmented to construct a region adjacency graph (RAG), which captures the containment hierarchy of regions: which black regions are inside which white regions and vice versa. This graph forms a rooted tree structure. A canonical form for unordered rooted trees is computed by recursively ordering children according to the lexicographic weight of their depth sequences. Each fiducial in the marker set has a unique topological structure, so its canonical name can be looked up in a dictionary to identify the marker (Bencina et al. 2005b).

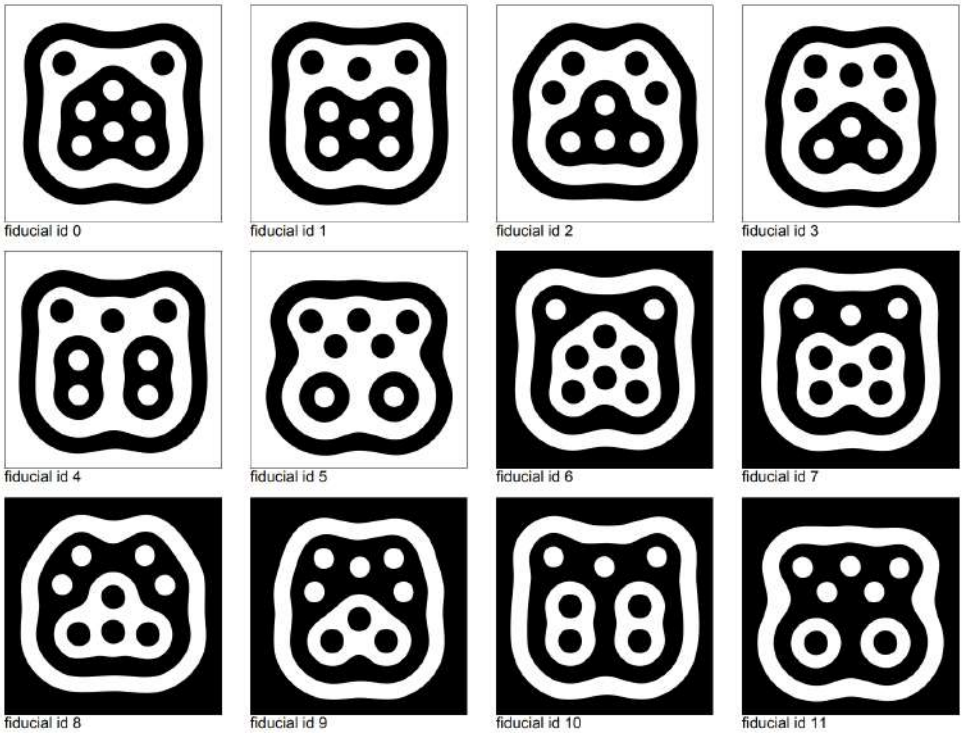


Figure 2.7: reacTIVision Amoeba symbols. CC BY-NC-SA 2005 Martin Kaltenbrunner <martin@tuio.org> and Ross Bencina <rossb@audiomulch.com>

Fiducial location and orientation are computed directly from the bounding boxes and

centroids of the tree's leaf nodes, without requiring separate corner detection or line fitting. This makes the method robust to scale changes and viewing angle variations, since the topology is preserved under affine transformation. This system supports sets of hundreds of unique fiducials and achieves real-time performance (Bencina et al. 2005b). Fiducials generated for reacTIVision are significantly smaller than prior systems, owing to the use of topologically optimized circular leaf regions rather than simple geometric shapes.

2.3.9 Deep Learning-Based Markers: E2ETag and DeepFormableTag

Recent advances have applied deep learning to fiducial marker design and detection. E2ETag is an end-to-end trainable method that jointly optimizes marker appearance and detection network parameters (Peace et al. 2021). Rather than using hand-crafted patterns or codes, the marker generator network learns to create visual codes that are simultaneously easy for the detector network to recognize and robust to common distortions. However, early deep learning approaches had reliable performance with only 30 distinct markers.

Figure 2.8: E2ETag example markers. Image from Peace et al. (2021), used under the MIT License.

DeepFormableTag extended learning-based markers to handle severely deformable surfaces (Yaldiz et al. 2021). Traditional markers require the marker to be attached to a planar surface. When the surface deforms (as in motion capture or structured light on dynamic surfaces), markers become warped and corners shift, causing traditional detectors to fail. DeepFormableTag consists of three components: a learnable marker generator creating unique free form color patterns, a differentiable photorealistic renderer that simulates markers on deformed surfaces, and a deformation-aware detector that

recognizes and decodes markers via inverse deformation transformation. The end-to-end optimization allows the system to learn markers that remain decodable under severe shape deformation (Yaldiz et al. 2021). This method significantly outperforms traditional and prior learning-based systems on metrics including detection rate, accuracy, and robustness to adverse imaging conditions.

2.3.10 Position-Sensing Markers: De Bruijn Tori and Marker Fields

A fundamentally different approach addresses scenarios with extremely limited camera field of view or severe visibility problems: position-sensing markers seamlessly tile an entire surface, encoding both position and identity in every local region.

A de Bruijn sequence is a 1D binary string where every possible length-n substring is unique. A de Bruijn torus is the 2D generalization: a binary matrix where every $m \times n$ submatrix is unique (Lynch et al. 2025). For example, in digital paper (Anoto system), dots are printed with slight offsets encoding a 4-state matrix, constructed from the cross of two 6-order de Bruijn sequences. A camera beneath the paper observes a 6×6 local region, which uniquely identifies the camera's absolute position. However, de Bruijn tori suffer from a critical limitation: submatrices are unique only in one orientation, so markers lack orientability (Lynch et al. 2025).

Marker fields address this limitation. An n-order marker field is a binary matrix where every $m \times n$ subregion is unique in all four rotations (0°, 90°, 180°, 270°). This orientable uniqueness property is essential: when a camera observes a local region, the system can not only determine position but also orientation without additional sensors. Position-sensing markers composed from marker fields enable reliable localization even when the field of view captures only partial ID tags, because the seamless tiling ensures that every possible observation maps to a unique position and orientation pair (Szentandrasi et al. 2012).

2.3.11 HydraMarker: Advanced Marker Fields for Position-Sensing

Generating marker fields is computationally intensive, limiting their practical accessibility. Earlier methods relied on brute force search restricted to small fields (e.g., 3-order 12×12) (Szentandrasi et al. 2012), genetic algorithms run on supercomputers (Szentandrasi et al. 2012), or accelerated brute force with precomputed tag indices achieving 4-order fields up

to approximately 20×20 (Gao et al. 2021). The fundamental bottleneck is satisfying three constraints simultaneously: no self-conflict (ID tags cannot be identical to themselves after rotation), no cross-conflict (ID tags cannot match others under any rotation), and no seams (ID tags must exist in every subregion from every orientation) (Szentandrasi et al. 2012). These computational limitations prevented marker fields from becoming practical for real world applications. HydraMarker solves the marker field generation problem through a novel approach inspired by wave function collapse (WFC) algorithms (Zhu et al. 2023). Rather than exhaustive search or genetic algorithms, it formulates marker field generation as a constraint satisfaction problem and applies an intelligent greedy strategy to solve it with dramatically improved efficiency, making marker field generation practical for common users on standard laptops.

The core innovation is the bWFC (binary wave function collapse) algorithm (Zhu et al. 2023). This method minimizes the probability of wrong assignments by always collapsing the most uncertain or most constrained unknown element. This system introduces two key metrics: a likelihood matrix L indicating whether 0 and 1 are balanced (equiprobable) at each position within the unassigned regions, and a support matrix C measuring how many remaining valid ID tags support each candidate assignment (0 or 1) at each position. At each iteration, the algorithm identifies the position with the lowest support value (most constrained), and assigns its state to match the likelihood preference: if the position has balanced likelihood but minimal support for 0, assignment of 0 is prioritized to maintain larger remaining codebook. This greedy selection of the most constrained element, combined with intelligent backtracking when conflicts arise, enables far more efficient generation than exhaustive search while maintaining constraint satisfaction (Zhu et al. 2023). Additionally, HydraMarker introduces fast-bWFC, an optimized variant that replaces full superstate constraint checking with lightweight risk metrics reducing computational overhead while maintaining solution quality (Zhu et al. 2023). Empirically, fast-bWFC generates a 4-order 85×85 marker field in approximately 1.5 minutes on a laptop, whereas prior methods require weeks of computation or supercomputer resources (Zhu et al. 2023).

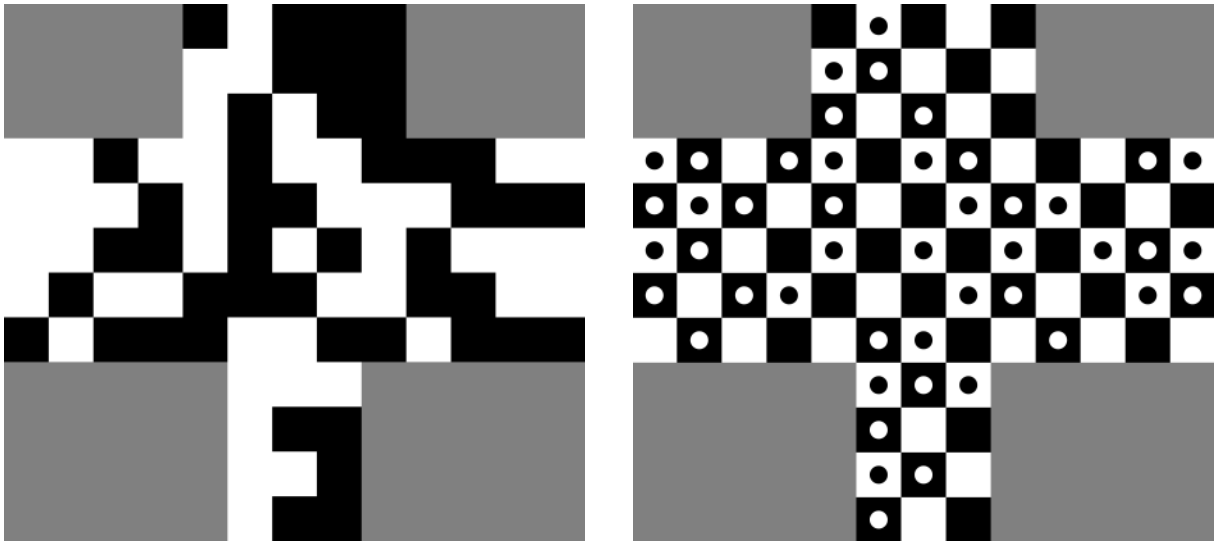
HydraMarker broadens the definition of marker fields, enabling unprecedented flexibility beyond traditional rectangular fields with square ID tags. First, advanced marker field definition allows arbitrary field shapes and supports preset values: specific matrix

elements can be fixed to embed required patterns within the field or to constrain field boundaries, enabling non-rectangular geometries such as circular or polygonal marker fields tailored to specific applications (Zhu et al. 2023). Second, multifold local uniqueness represents a paradigm shift in position-sensing markers. Rather than enforcing a single tag shape size, HydraMarker simultaneously supports multiple tag shapes (e.g., 4×4, 3×6, 2×9, 1×20) within the same marker field, where each tag shape maintains local uniqueness across all four rotations. This dramatically improves robustness to occlusion: under extreme occlusion where one tag size is completely obscured, a different tag size might still be visible and readable, providing multiple fallback decoding paths. The tradeoff between small tags (which fit more position identifiers per unit area but are harder to detect reliably) and large tags (which are easily detected but encode fewer positions) is thereby eliminated. Third, practical accessibility makes marker field generation a commodity tool rather than a research artifact. While previous methods required supercomputer resources due to prohibitive computational cost, HydraMarker's fast-bWFC generates marker fields on standard laptops, making marker field technology practical for real-world applications (Zhu et al. 2023).

HydraMarker's experimental results demonstrate superior error tolerance and quality compared to prior methods (Zhu et al. 2023). For 4-order marker fields with a 4×4 local observation window, HydraMarker achieves inter-window confusion possibility (probability that a local 4×4 observation is incorrectly decoded as a different position) of approximately 0.03% at 5% bit error rate, significantly lower than simpler marker field designs from Szentandrasi et al. (2012) (Zhu et al. 2023). Detection and identification of position-sensing markers proceed by applying standard fiducial detection algorithms (edge-based line fitting, quad assembly, homography computation) to extract local bit patterns, then matching these patterns against the marker field codebook to simultaneously decode position and orientation (Zhu et al. 2023). Error tolerance arises from inherent redundancy: multiple overlapping local regions are typically observed in a camera's field-of-view, and their decoded positions should be consistent; inconsistencies indicate bit errors that can be detected and reported to higher-level algorithms.

As an illustrative example, a cross-shaped 12×13 marker field supporting 3×3, 5×5, and 1×13 ID tags simultaneously is generated, as shown in Fig. 2.9 (a). This abstract marker field is then instantiated on a printable board by overlaying it onto a

checkerboard pattern: wherever the marker field contains a white cell, a circular dot is placed with opposite color relative to the underlying checkerboard square. The resulting dot pattern directly corresponds to a binary state matrix whose entries assume values of 1 and 0, indicating dot absence and presence, respectively. The resulting physical marker, combining the multifold locally unique tag structure with a checkered background, is shown in Fig. 2.9 (b). Gray cells in Fig. 2.9 (a) represent locked cells, which are removed in the final printable design to obtain the cross-shaped marker.



(a) (b)

Figure 2.9: (a) Cross-shaped marker field supporting 3×3, 5×5, and 1×13 ID tags simultaneously. (b) Final marker obtained by instantiating the marker field in (a) on a checkered background.

From a sensing standpoint, more sophisticated fiducial families (e.g. lexicode-based square grids, dense ChArUco boards, cyclic-code markers, and position-sensing marker fields) do not intrinsically require different camera technologies than simpler early markers. However, for a fixed working distance and field of view, their higher information density imposes a minimum number of image pixels per marker cell or local pattern, so maintaining comparable detection and pose-estimation accuracy is achieved either by using higher-resolution cameras or by increasing the physical marker size in the scene.

Chapter 3

Materials: System Design and Development

3.1 Hardware Configuration

3.1.1 Basler Dart Camera and Lens Assembly

The vision system employed a Basler daA1920-160uc camera, a compact 2.3 MP industrial camera selected in a previous study for its minimal form factor ($20\,\mathrm{mm} \times 20\,\mathrm{mm} \times 29\,\mathrm{mm}$, $<15\,\mathrm{g}$), affordable cost, and compatibility with the Basler pylon Software Suite API (Basler AG 2024). The camera features a $1/2.3''$ Sony IMX392LQR-C CMOS global shutter sensor with a native resolution of 1936×1216 pixels and a maximum frame rate of 67 fps at default settings, expandable to 164 fps when throughput limiting is disabled (Basler AG 2024).

The optical subsystem comprised an Edmund Optics #33-300 TECHSPEC® UC Series lens with a $4\,\mathrm{mm}$ fixed focal length (Edmund Optics Inc. 2024). This lens is optimized for $1/2.5''$ sensors and provides a variable aperture range of f/1.8 to f/11, enabling flexible illumination control across diverse clinical environments (Edmund Optics Inc. 2024). The lens specifications include a compact physical footprint ($40.6\,\mathrm{mm}$ length, $40\,\mathrm{mm}$ maximum diameter, $72\,\mathrm{g}$), a maximum image circle of $8\,\mathrm{mm}$, and $< 17.5\%$ distortion on the camera's sensor format, delivering high optical resolution (> 200 lp/mm across the usable image circle for $2.2\,\mu\mathrm{m}$ pixels) (Edmund Optics Inc. 2024).

These specifications are particularly advantageous for tracking applications requiring

fast acquisition rates and high optical resolution, while maintaining a lightweight form factor suitable for the previous Inside-Out tracking approach where the camera and lens assembly is integrated with the ultrasound probe.

For experimental validation, the optical assembly was mounted on a stable laboratory tripod as illustrated in Fig. 3.1. However, the final clinical Outside-In implementation envisions integration onto a commercial articulated mounting arm equipped with quick release clamps to secure the system to the structural tubing on patient bed frames. This approach maximizes flexibility for intraoperative positioning while minimizing workspace intrusion.

3.1.2 2D Ultrasound Probe (Clarius HD3 L7)

The ultrasound imaging component was a Clarius HD3 L7 wireless linear array transducer (Clarius Medical 2024), shown in Fig. 3.1. This platform was selected for its inherent wireless capability, enabling direct streaming of ultrasound video over WiFi to a tablet or desktop workstation without tethered connections. The L7 operates across a frequency range of 4–13 MHz with a maximum imaging depth of 15 cm, making it well-suited for superficial anatomical structures (Clarius Medical 2024). Physical specifications include compact dimensions ($147\,\mathrm{mm} \times 76\,\mathrm{mm} \times 32\,\mathrm{mm}$) and a lightweight design ($288\,\mathrm{g}$), facilitating ergonomic handling during extended imaging sessions (Clarius Medical 2024). The device is IP67-rated, permitting full submersion for sterilization using approved disinfectants, thereby supporting clinical workflow requirements.

Figure 3.1: 2D Ultrasound Clarius L7 probe (left). Basler daA1920-160uc and Edmund Optics Lens assembly (right).

3.2 Proposed Dual HydraMarker Approach

The proposed Outside-In approach mounted the Basler Dart camera approximately $300\,\mathrm{mm}$ from the scanning zone, viewing the ultrasound probe from the side opposite to the sonographer. This configuration provided a large field of view, allowing the sonographer a wide range of motion during scanning without losing line of sight to the camera.

3.2.1 HydraMarker on Ultrasound Probe

The HydraMarker pattern attached to the ultrasound probe was first designed to maximize the usable patterned area. The initial concept was a wrap-around marker covering both the front and side surfaces of the probe. However, practical constraints, such as printing the marker on paper and the presence of sharp edges on the probe prototype, prevented a clean wrap and made this design infeasible. The pattern was therefore modified to a cross-shaped configuration that wraps only around the central portion of the sides, leaving the upper and lower edges uncovered due to their sharpness and incompatibility

with paper-based wrapping. This geometry was obtained by locking selected cells in the fast-bWFC algorithm so that the generated marker field matched the desired shape (see Fig. 2.9).

3.2.2 HydraMarker on Patient

In addition to the fiducial marker on the ultrasound probe, the system was designed to accommodate a second HydraMarker attached to the patient to monitor patient motion during the scan. By continuously tracking this marker, the system can detect patient movements exceeding a predefined threshold and flag acquisitions that may be affected by motion artifacts, prompting the sonographer to repeat or discard compromised scans.

The current patient-marker concept was tailored to the clinical workflow of the Graf scan. During acquisition the newborn is positioned laterally, and the device is placed on the lateral hip region. Placement should follow recognizable anatomical landmarks, with the iliac crest considered a primary reference. The design, illustrated in Fig. 3.2, was conceived as an annular support featuring a central rectangular slit that defines the scanning window. The slit was aligned with the expected sweep direction and was sized to match the acquisition protocol: a length of approximately $50\,\mathrm{mm}$ and a width comparable to the probe transducer (approximately $40\,\mathrm{mm}$). Beyond maintaining access to the scanning region, this opening provided a visual and physical guide for the intended sweep length, addressing the observed tendency of experienced sonographers to reduce the sweep extent once the Graf plane is found, by scanning locally around that plane rather than completing a full $50\,\mathrm{mm}$ trajectory.

3.3 Marker Design Optimization Process

The design of fiducial markers for optical tracking involved balancing competing engineering objectives: maximizing the number of unique identifiable features (for redundancy and tracking robustness), ensuring those features are detectable given camera optics and working distance constraints, and maintaining physical practicality (printability and physical size). In the paediatric setting, an additional critical constraint is anatomical compatibility: marker dimensions must remain small relative to the available bony surface on the infant pelvis, so that the marker can be securely attached without obstructing

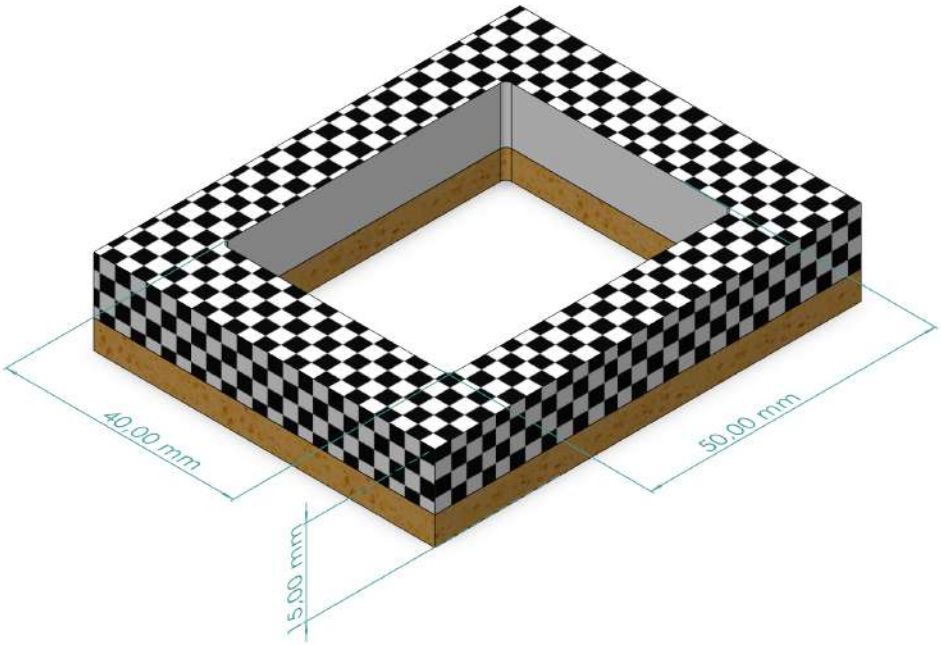


Figure 3.2: Prototype of the patient marker support featuring a checkerboard surface pattern (HydraMarker) and a silicone foam bottom layer.

access to anatomical landmarks or interfering with clinical handling of the patient.

3.3.1 Problem Overview and Motivation

Marker-based optical tracking relies on detecting and localizing distinctive fiducial features (in this case, corner points) within the camera's field of view. The ability to robustly detect these features depends on several interconnected factors:

- **Optical projection:** How the physical marker is projected onto the camera sensor as a function of working distance ($250\,\mathrm{mm}$ to $350\,\mathrm{mm}$) and lens focal length ($4\,\mathrm{mm}$);
- **Feature detectability:** Whether features are large enough (in pixels) to survive Gaussian smoothing and corner detection algorithms without being blurred or suppressed as noise;
- **Grid density:** How many identifiable features can fit within the physical marker area while remaining detectable across the full operating envelope;
- **Manufacturing constraints:** Whether the printed marker can be produced with sufficient precision and feature resolution (minimum feature size $\geq 0.06\,\mathrm{mm}$ achievable by the Xerox Altalink C8055 printer at 1200 DPI).

The optimization problem thus requires defining design variables (Gaussian smoothing parameter σ, corner detection patch size r, grid geometry), hard constraints (marker dimensions, sensor area limits, printer resolution, minimum corner count), and objective functions. Multiple valid solutions exist, each representing a different design philosophy: maximizing identification capacity (N_{IDs}), maximizing detection robustness ($s_{\min,\text{pixels}}$), or achieving balanced compromise between the two objectives.

3.3.2 Optimization Framework and Design Selection

A systematic multi-objective optimization framework was employed to search the design space and identify candidate marker configurations. The framework evaluated Gaussian smoothing parameters ($\sigma \in [0.6, 3.0]$ and corner detection patch sizes $r \in [1, 12]$, yielding patch dimensions from 3×3 to 25×25 pixels) across both square and rectangular grid geometries.

The search was constrained by three critical requirements: the marker projection onto the 1936 $\times$ 1216 sensor must not exceed 25% of the image area at the closest working distance ($250\,\text{mm}$), ensuring the sonographer maintains a wide field of view and range of motion; all printed corner features must exceed the minimum printer resolution ($0.06\,\text{mm}$) and remain visually detectable at the farthest working distance ($350\,\text{mm}$) after Gaussian smoothing; the design had to support at least 45 unique identifiable corners to provide redundancy for robust tracking under partial occlusion.

Three optimization objectives were evaluated:

- **Maximum Identification Capacity:** Prioritized maximizing N_{IDs} to pack as many detectable features as possible into the marker area, yielding higher grid density but potentially reduced robustness under noise or poor illumination.

- **Maximum Robustness:** Prioritized maximizing the minimum pixel size ($s_{\min,\text{pixels}}$) to ensure high corner detection reliability under challenging conditions (poor contrast, motion blur, variable lighting), at the cost of reduced identification capacity.

- **Balanced Compromise:** Seeks a middle ground between capacity and robustness via L^2 Euclidean distance to a utopia point, selecting designs that excel in both objectives rather than extremes that maximize only one.

The optimization algorithm performed an exhaustive parameter sweep, evaluating all feasible combinations of σ and r subject to the four constraints above. Infeasible configurations (violating any constraint) were excluded. For each candidate, the three objective metrics were computed, candidates were then ranked according to the selected optimization objective. For the balanced selection, the two key objectives (identification capacity and robustness) were normalized to $[0, 1]$ by scaling each metric relative to its best achievable value across the feasible set. This produces a normalized two-objective space in which $(1, 1)$ represents the ideal (simultaneously maximal) configuration. The selected design is the candidate with minimum Euclidean distance to $(1, 1)$, referred to as the utopia point. To visualize the outcome of the parameter sweep, Figures 3.3 – 3.4 summarize all feasible configurations in the normalized plane.

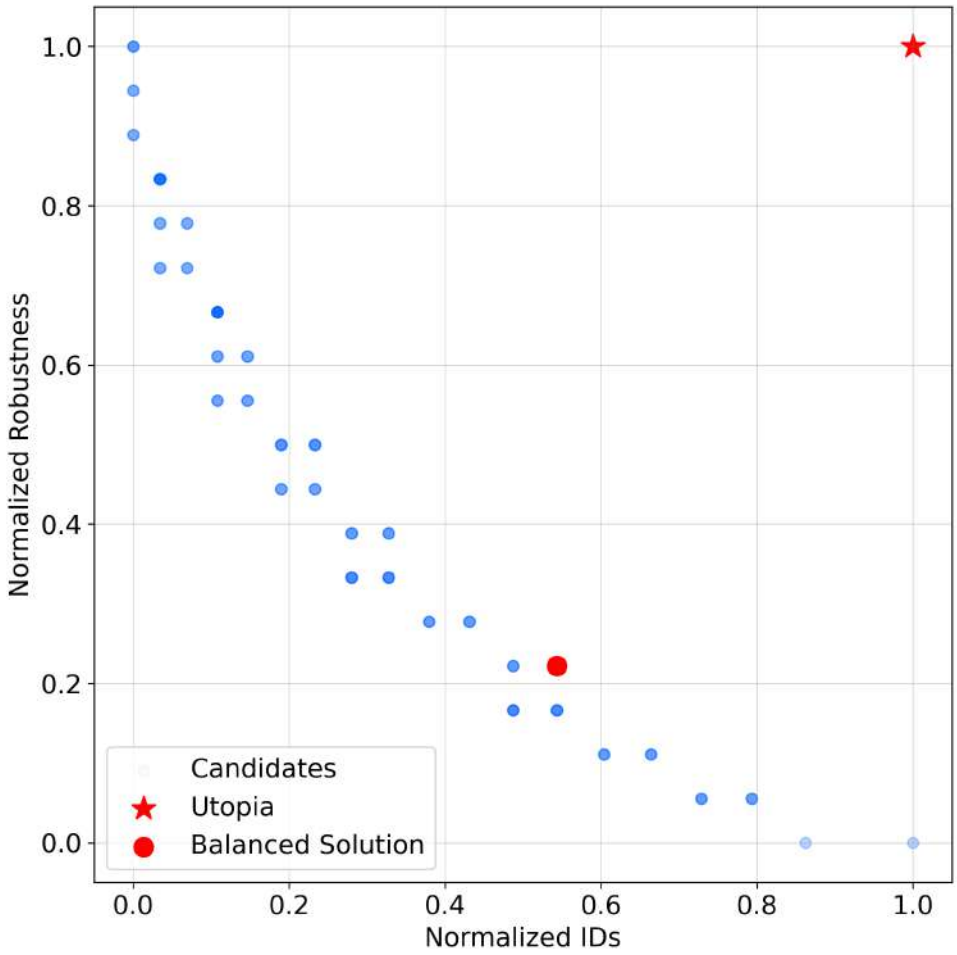


Figure 3.3: Feasible marker configurations in the normalized objective space. The balanced configuration selected for the probe marker is highlighted in red and the ideal target point is shown as a star for reference.

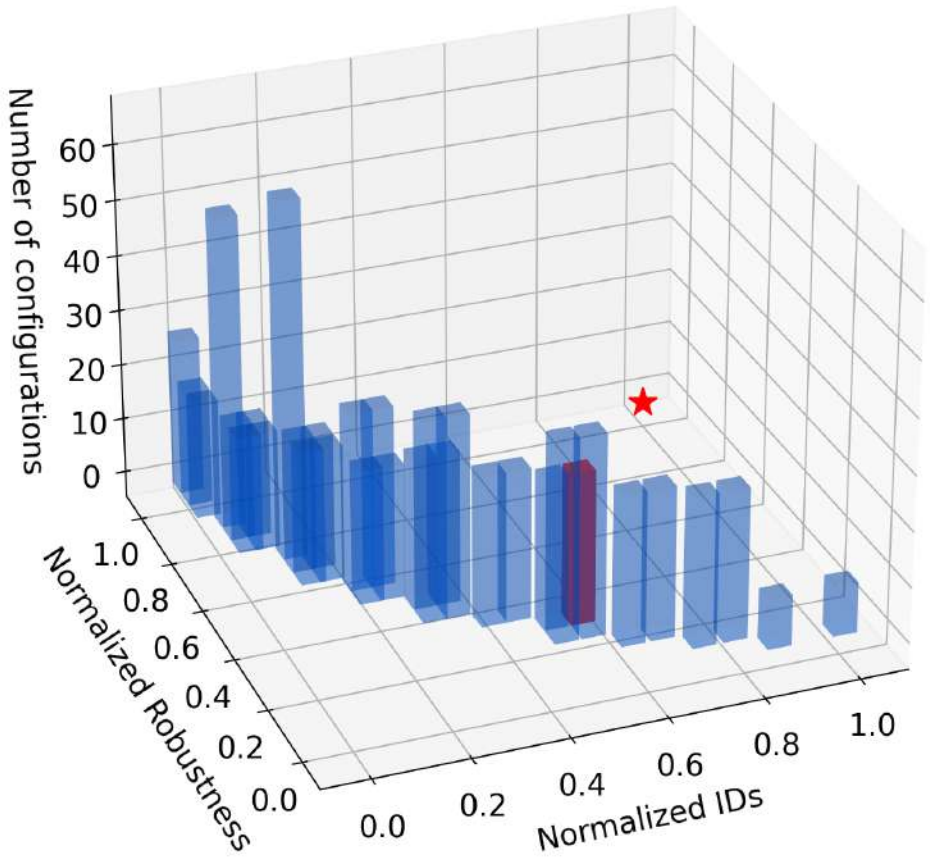


Figure 3.4: Histogram of feasible marker configurations in the same normalized objective space as Fig. 3.3. Bar height indicates the number of configurations and the balanced configuration is highlighted in red. The utopia point is indicated by a star.

The balanced compromise solution (12×13 marker field) was initially selected for the marker on the probe, because it provides the best trade-off between identification capacity and detection robustness. In its full rectangular layout, this design provided 182 uniquely identifiable corners within a $120\,\mathrm{mm} \times 130\,\mathrm{mm}$ area. To match the probe prototype geometry (see Fig. 2.9), the layout was subsequently trimmed to the usable region, reducing the total number of corners to 118. Preliminary experiments using this design were conducted to validate the Outside-In configuration and assess tracking performance. Based on the results, further design iterations were performed to enhance robustness and accuracy as described in Chapter 5.

For the marker attached to the patient, a maximum-ID solution was selected due to the limited available surface area. The resulting marker field, shown in Fig. 3.5, consisted of a 7×9 layout with 2×3 ID tags. In the original full rectangular configuration, the marker contained 80 corners; after removing the gray cells, the remaining number of detectable corners was therefore 65. This marker was intended to be mounted on the

support shown in Fig. 3.2.

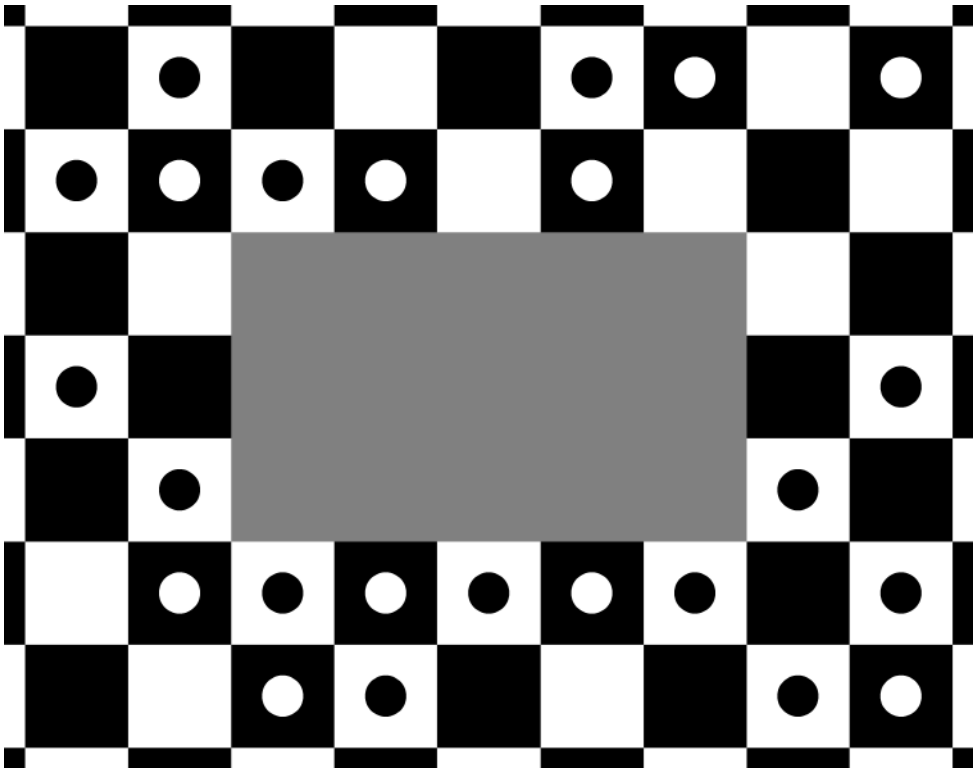


Figure 3.5: Annular-shaped marker field supporting 2×3 ID tags to be attached to the patient support.

Detailed mathematical formulations, constraint definitions, objective function definitions, design variable ranges, the parameter sweep algorithm, optimization outputs, and ranking criteria are provided in Appendix A.

3.4 Marker Detection and Identification

The marker detection and identification algorithm located fiducial corners on the printed marker and assigned unique identifiers based on their grid position. The implementation was based on the HydraMarker framework (Zhu et al. 2023), originally developed in C++ and was translated to Python in this work for integration with the broader ultrasound tracking pipeline. Significant performance optimizations were applied while preserving the original algorithmic accuracy.

3.4.1 Image Acquisition and Preprocessing

Grayscale images were acquired from the Basler daA1920-160uc camera at the configured resolution (1936×1216 pixels). The image was converted to 32-bit floating-point format to preserve gradient information and enable sub-pixel refinement. No additional preprocessing was applied, allowing the detection algorithm to work directly on raw camera output. All experiments were conducted under standard laboratory ambient illumination without dedicated lighting control or intensity normalization, so the reported performance reflects operation in a generic, non-optimised lighting environment.

3.4.2 Stage 1: Corner Detection via Gradient Analysis

The corner detection stage identified candidate corner locations by analyzing local image gradients. The algorithm computed:

1. Gradient computation: Sobel operators compute g_x and g_y (gradients in x and y directions) using a 3×3 kernel with reflection boundary handling;

2. Magnitude and smoothing: the gradient magnitude was computed as:

$$M(\mathbf{r}) = \sqrt{g_x^2(\mathbf{r}) + g_y^2(\mathbf{r})} \tag{3.1}$$

   This was then smoothed with a Gaussian kernel of standard deviation σ:

$$M_{\text{smooth}}(\mathbf{r}) = G_\sigma * M(\mathbf{r}) \tag{3.2}$$

3. Corner response: this was defined as the difference between the smoothed magnitude and the original magnitude:

$$R(\mathbf{r}) = M_{\text{smooth}}(\mathbf{r}) - M(\mathbf{r}) \tag{3.3}$$

4. Non-maximum suppression: a 3×3 morphological dilation identified local maxima. Pixels where the dilation exceeds the original response value were suppressed, leaving only peak responses.

5. Thresholding and selection: response values below a threshold were discarded. The remaining candidates were ranked by response strength, and the top N_{expect} corners were selected. The expected count is:

$$N_{\text{expect}} = 10 \times (G_x + 1)(G_y + 1) \tag{3.4}$$

where G_x and G_y are the grid dimensions.

The output was an unordered set of corner locations in pixel coordinates, typically numbering in the hundreds to low thousands depending on image contrast and noise.

3.4.3 Stage 2: Corner Refinement and Feature Extraction

Detected corners were refined to sub-pixel accuracy and were characterized by local edge orientation. This stage operated on each candidate corner independently.

For each corner at integer pixel location $(\mathbf{r}_0)$, a neighborhood was extracted and gradients are fit to a 2D surface model. The refinement iterated twice:

1. Extract the $(2r + 1) \times (2r + 1)$ gradient neighborhood g_x, g_y around the corner.
2. Model the corner as a position where the weighted sum of gradients is minimized:

$$\text{minimize} \quad \sum_{(i,j)} \left[(g_y[i,j] \cdot (i - r_y) + g_x[i,j] \cdot (j - r_x))^2 \right] \tag{3.5}$$

3. Solve the normal equations using a 2×2 system derived from the Hessian of this cost function;
4. Update the corner position to the solution $(\mathbf{r}_{\text{refined}})$;
5. Repeat for a second iteration to improve convergence.

The refinement increased localization accuracy from ± 0.5 pixels (detector grid) to < 0.1 pixels.

For each refined corner, local intensity was sampled in a $(2r+1)\times(2r+1)$ neighborhood and fit to a 2D quadratic polynomial:

$$I(x, y) \approx ax^2 + bxy + cy^2 + dx + ey + f \tag{3.6}$$

The gradient of this polynomial defined two principal edge directions (roots of the associated conic). The edge orientations were extracted as:

$$\theta_0, \theta_1 = \arctan(\text{roots of } \partial_y I / \partial_x I) \tag{3.7}$$

These two angles defined the "ledge" (local edge orientation pair) at each corner. A quality score was computed by:

- Correlation between the polynomial model and the sampled patch (should be high for well-formed corners);
- Angle bias: deviation from orthogonality ($|\theta_0 - \theta_1| \approx 90^\circ$ is required for marker corners);
- Hierarchical clustering using Chebyshev distance to merge nearby corners with similar ledges

Corners with low quality scores or angle bias $> 20^\circ$ were discarded. This filtering typically reduced the candidate set from hundreds to tens of corners.

3.4.4 Stage 3: Grid Structure Building

The refined corners were organized into grid connectivity using Delaunay triangulation and compass based grid assignment.

1. Delaunay triangulation: compute the Delaunay triangulation of the refined corner positions (in (x, y) coordinates);
2. Extract edges: from the triangulation simplices, extract all unique edges connecting pairs of corners;
3. Ledge-based filtering: for each edge connecting corners i and j, validate that the edge direction aligned with both corners' ledge angles (within $\pm 30^\circ$). Edges that fail this check were discarded, as they likely connect non-adjacent grid cells or noise;
4. Length filtering: sort remaining edges by length. Discard any edge longer than $1.7\times$ the shortest edge in its neighborhood, removing spurious long-range connections.

Using the validated edge set, a grid connectivity graph was built by tracing edges and assigning grid coordinates (i, j) to each corner:

1. Maintain a "compass" orientation at each corner, tracking which direction (up, down, left, right) each edge points;
2. Starting from an unvisited corner, perform breadth-first search along edges, incrementing/decrementing grid coordinates (i, j) as the search progresses;
3. Each connected component formed one grid. Multiple grids in a single image indicate multiple markers or detection artifacts.

The output was a set of 2D grid matrices, where each cell contains the global index of the corner at that grid position (or NaN if missing).

3.4.5 Stage 4: Marker Identification

The final stage assigned unique identifiers to corners by matching the detected grid structure against the known marker field (stored in the state matrix, described in the subsection 2.3.11).

For each grid, a "dot table" was constructed by examining 2×2 blocks of corners. The intersection of four adjacent corners represented the center of a printed dot (if all four corners are detected):

$$\text{dot\_center} = \frac{1}{4} \sum_{i=0}^{1} \sum_{j=0}^{1} \mathbf{r}[i, j] \tag{3.8}$$

At each dot location, the pixel intensity was sampled and compared against a threshold to determine whether a dot is present or absent. This binary dot pattern was encoded into the dot table.

The detected dot table was matched against the known sta pattern (subsection 2.3.11) using 2D convolution:

$$\text{match\_score} = \sum_{(i,j)} \text{sta}[i, j] \times \text{dot\_table}[i, j] \tag{3.9}$$

A high match score indicated correct alignment. The matching also detected the position and orientation of the detected grid relative to the state, enabling ID assignment

based on grid position.

The final output was a list of corner coordinates (in image space) paired with their assigned IDs.

3.4.6 Performance and Computational Complexity

The detection pipeline was optimized through several targeted improvements to address bottlenecks identified via profiling:

The initial implementation translated the C++ HydraMarker code directly, resulting in per-frame processing times of approximately 1200 ms on a single CPU core. Profiling revealed that corner detection consumed 97.6% of total time, primarily due to repeated gradient computation and non-vectorized operations.

Key optimizations applied:

1. OpenCV acceleration: replaced pure NumPy gradient and smoothing with OpenCV (Sobel, GaussianBlur, dilate), which are highly optimized C++ implementations. All operations are now performed in a single pass per frame.

2. Matrix caching: the polynomial fitting step reuses a pre-computed design matrix A (computed once per radius r) rather than recomputing it for each of the hundreds of corner candidates;

3. Vectorized operations: patch extraction using cv2.getRectSubPix processes all candidates in a tight loop; correlation and angle bias filtering are applied to stacked arrays rather than per-corner loops;

4. Direct linear solve: refinement uses a cached 2×2 normal system solution instead of calling np.linalg.lstsq (general least-squares solver) per corner;

5. Lightweight clustering: hierarchical clustering for point merging was replaced with a simple Chebyshev-distance bucket union-find, avoiding the overhead of SciPy linkage.

The fully optimized version achieved approximately $\mathbf{6}\times$ speedup compared to the initial translation.

Despite aggressive optimization, corner detection and refinement remained the primary computational burden, suggesting that further speedup would require algorithmic changes.

3.5 3D Marker Reconstruction

The marker detection stage produced a set of 2D image observations (IDs positions in pixel coordinates across multiple frames). To enable 3D pose estimation and surface tracking, these 2D observations had to be lifted into 3D space and registered into a common coordinate frame. This section describes the structure-from-motion (SfM) pipeline that reconstructed the 3D positions of all IDs, estimated the camera pose (position and orientation) for each frame, and produced a metric 3D map aligned to the object coordinate system.

The pipeline consisted of five stages: detection preprocessing to filter inconsistent observations, two-view bootstrap initialization to establish the initial 3D structure and scale, incremental pose estimation to register remaining frames and triangulate new IDs, bundle adjustment to refine all camera poses and IDs positions simultaneously, and metric scaling and coordinate frame alignment to obtain a final object centric map.

3.5.1 Detection Preprocessing and Filtering

Before attempting to reconstruct 3D structure, the raw IDs detections were filtered to remove inconsistent observations. This preprocessing stage removed frames and IDs that are outliers, noise, or inconsistent with the global detection consensus. The motivation was to provide bundle adjustment with a good starting point rather than a set with large outliers that could cause the nonlinear optimizer to converge to a poor local minimum.

Both frame and epipolar consistency filtering were applied.

For each marker ID that appears across multiple frames, the detector compared the IDs' position consistency. Frames where IDs deviate significantly from the consensus position (beyond $n\sigma$ threshold, where σ is the standard deviation of positions across frames, and n is a configurable multiplier) were flagged as outliers. Additionally, motion estimates are computed: if an ID exhibits a sudden velocity spike (e.g., jumping more than 80$\times$ the median per-frame motion), the frame was discarded.

IDs that are geometrically inconsistent across frame pairs were identified using epipolar geometry. For each pair of frames sharing sufficient ID observations, an essential matrix was estimated using RANSAC. IDs that consistently fail the epipolar constraint (i.e., are inliers in $< 80\%$ of frame pairs where they appear) were removed from the dataset.

The result of preprocessing was a filtered set of frames and IDs that form a more coherent detection set, reducing the chance of bundle adjustment becoming trapped in a poor local minimum.

3.5.2 Bootstrap Initialization

The 3D reconstruction was initialized using two-view geometry. A frame pair was automatically selected based on shared IDs count, inlier ratio, and baseline separation (parallax). The two-view geometry problem was solved as follows:

Given two frames $\mathbf{F}_1$ and $\mathbf{F}_2$ with shared IDs observations at normalized (undistorted) image coordinates, the essential matrix $\mathbf{E}$ was estimated:

$$\mathbf{p}_2^T \mathbf{E} \mathbf{p}_1 = 0 \tag{3.10}$$

where $\mathbf{p}_1, \mathbf{p}_2$ are homogeneous coordinates of corresponding IDs observations. The essential matrix encodes the relative pose (rotation $\mathbf{R}$ and translation direction $\mathbf{t}$, up to scale) between the two camera centers.

The essential matrix was recovered from the fundamental matrix using RANSAC to reject outliers, then decomposed via singular value decomposition:

$$\mathbf{E} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T \tag{3.11}$$

This decomposition yielded two possible rotation matrices and the translation direction. The correct solution was selected by checking that triangulated IDs have positive depth in both cameras.

Once the relative pose is known, 3D IDs positions were recovered by linear triangulation. For each ID correspondence, the two normalized image rays were intersected in 3D space:

$$\mathbf{p}_3^{\text{homog}} = \mathbf{P}_1 \mathbf{X} \quad \text{and} \quad \mathbf{p}_3^{\text{homog}} = \mathbf{P}_2 \mathbf{X} \tag{3.12}$$

where $\mathbf{P}_1 = [\mathbf{I}, \mathbf{0}]$ and $\mathbf{P}_2 = [\mathbf{R}, \mathbf{t}]$ are the projection matrices for the two cameras, and $\mathbf{X}$ is the homogeneous 3D point. The point was solved via linear least-squares:

$$\mathbf{X} = \arg\min_{\mathbf{X}} \left\| \begin{bmatrix} \tilde{\mathbf{p}}_1 \mathbf{P}_1 \\ \tilde{\mathbf{p}}_2 \mathbf{P}_2 \end{bmatrix} \mathbf{X} \right\|^2 \tag{3.13}$$

where $\tilde{\mathbf{p}}$ is the cross-product matrix. Validity checks ensured positive depths in both cameras and reprojection error below a threshold. Valid triangulated IDs formed the initial 3D point cloud and seed the reconstruction.

If validation of the initial reconstruction failed, an alternative frame pair was attempted. If all candidate pairs failed, bootstrap was aborted and the user was alerted to check data quality.

3.5.3 Incremental Pose Estimation

After bootstrap, the remaining frames were registered one-by-one using the PnP (Perspective-n-Point) algorithm, and new IDs were triangulated as they become visible.

For each unregistered frame $\mathbf{F}_k$, the camera pose (rotation $\mathbf{R}_k$ and translation $\mathbf{t}_k$) was estimated using observed IDs already in the 3D map. Given a set of 3D-2D correspondences $\{(\mathbf{X}_i, \mathbf{p}_i)\}$, the PnP problem seeks to minimize:

$$\min_{\mathbf{R},\mathbf{t}} \sum_i \|\mathbf{p}_i - \text{proj}(\mathbf{R}\mathbf{X}_i + \mathbf{t})\|^2 \tag{3.14}$$

where proj is the camera projection function incorporating intrinsics and distortion. PnP was solved via RANSAC, followed by refinement using Levenberg-Marquardt optimization. Frames with insufficient inliers (< 6 IDs) were deferred to later iterations, allowing the map to grow incrementally.

Once a frame is registered, newly visible IDs (observed in the current frame but not yet in the map) were triangulated using the current frame and a reference frame from the set of registered frames. The triangulation follows the linear least-squares approach described in the bootstrap triangulation paragraph. New triangulated IDs were added to the map, enriching the point cloud for subsequent frame registration.

The incremental process continued until all frames are registered or no progress is made for a full iteration (indicating that remaining frames lack sufficient overlap with

the current map). A minimum of three frames was required before proceeding to bundle adjustment.

3.5.4 Bundle Adjustment

After all frames were registered, a global bundle adjustment step refined all camera poses and IDs positions simultaneously by minimizing the total reprojection error across all observations.

3.5.4.1 Optimization Problem Formulation

Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$ denote the 3D IDs positions and $\boldsymbol{\theta} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N\}$ denote the camera poses. Let $\mathbf{Z} = \{z_{ij}\}$ denote the set of observations (IDs i as observed in frame j). The bundle adjustment problem is:

$$\min_{\mathbf{X},\boldsymbol{\theta}} \sum_{(i,j)\in\mathbf{Z}} \left\| \mathbf{z}_{ij} - \text{proj}_j(\mathbf{R}_j\mathbf{x}_i + \mathbf{t}_j) \right\|^2 \tag{3.15}$$

where proj_j is the projection function for camera j (incorporating intrinsic parameters $\mathbf{K}$ and distortion coefficients $\mathbf{d}$). The cost function sums squared reprojection residuals over all observations.

To prevent rank deficiency (the scale and global translation are unobservable from image observations alone), the first camera was anchored: $\mathbf{R}_1 = \mathbf{I}$ and $\mathbf{t}_1 = \mathbf{0}$ are held fixed during optimization. This establishes an arbitrary reference frame; later reorientation and rescaling are performed in the post-processing stage.

Bundle adjustment was solved using the Ceres Solver, a C++ library for nonlinear least-squares optimization. Ceres uses a Levenberg-Marquardt method that adaptively adjusts the step size based on the agreement between the linear model (first-order Taylor expansion) and the actual objective function.

The solver constructs the Jacobian matrix $\mathbf{J}$, where each row corresponds to one residual and columns correspond to optimization variables. For efficiency, Ceres exploits the sparse block structure of the Jacobian in SfM problems: the Jacobian is approximately block-diagonal in camera/IDs variables, allowing the use of specialized sparse linear solvers.

The optimization step at iteration k solved:

$$\min_{\Delta \boldsymbol{x}} \|\mathbf{J}\Delta \boldsymbol{x} + \mathbf{r}\|^2 + \lambda \|\Delta \boldsymbol{x}\|^2 \tag{3.16}$$

where $\mathbf{r}$ is the residual vector, $\Delta \boldsymbol{x}$ is the step, and λ is a damping parameter. If the step reduced the objective, λ was decreased (moving toward Gauss-Newton); if the step increased the objective, λ was increased (moving toward gradient descent). This adaptive strategy ensured convergence even from poor initializations.

Convergence was determined by small gradient norms and small relative decreases in objective value. Typical convergence took 10–50 iterations depending on initialization quality (which is why preprocessing is important).

3.5.5 Scaling and Reorientation

The initial reconstruction is metric-ambiguous: the scale is arbitrary and the orientation is determined only by the choice of the first camera frame. To produce a final map in the object coordinate system with physical distances, two operations were performed:

3.5.5.1 Metric Scaling via Nearest-Neighbor Distance

The scale was estimated from the 3D IDs positions by computing nearest-neighbor distances. For each ID position $\mathbf{x}_i$, the distance to its nearest neighbor was computed:

$$d_i = \min_{j \neq i} \|\mathbf{x}_i - \mathbf{x}_j\| \tag{3.17}$$

The median of these nearest-neighbor distances was taken as an estimate of the mean grid spacing in the reconstruction:

$$d_{\text{median}} = \text{median}(\{d_1, d_2, \ldots, d_M\}) \tag{3.18}$$

The scale factor was then:

$$s = \frac{c_{\text{cell}}}{d_{\text{median}}} \tag{3.19}$$

where c_{cell} is the known physical spacing between adjacent cells in the grid. All 3D IDs positions and camera translations were scaled, whereas rotation matrices remained unchanged:

$$\mathbf{x}'_i = s \cdot \mathbf{x}_i, \quad \mathbf{t}'_j = s \cdot \mathbf{t}_j \tag{3.20}$$

The resulting reconstruction was now in physical units (mm).

3.5.5.2 Coordinate Frame Alignment

To align it to the object's coordinate system, a user-specified or automatically-derived custom coordinate frame was applied.

The user may optionally specify three marker IDs: one for the origin, and two others to define the X and Y axes. Given IDs positions $\mathbf{x}_{\text{origin}}$, $\mathbf{x}_X$, and $\mathbf{x}_Y$, a new coordinate frame is constructed using the Gram-Schmidt orthogonalization process:

1. Origin: $\mathbf{o} = \mathbf{x}_{\text{origin}}$
2. X-axis direction: compute the vector from origin to X-axis ID:

$$\mathbf{v}'_X = \mathbf{x}_X - \mathbf{o} \tag{3.21}$$

   Normalize to unit length:

$$\hat{\mathbf{x}} = \frac{\mathbf{v}'_X}{|\mathbf{v}'_X|} \tag{3.22}$$

3. Y-axis direction (pre-orthogonalization): compute the vector from origin to Y-axis ID:

$$\mathbf{v}'_Y = \mathbf{x}_Y - \mathbf{o} \tag{3.23}$$

4. Gram-Schmidt orthogonalization: remove the component of $\mathbf{v}'_Y$ parallel to $\hat{\mathbf{x}}$:

$$\mathbf{v}''_Y = \mathbf{v}'_Y - (\mathbf{v}'_Y \cdot \hat{\mathbf{x}})\hat{\mathbf{x}} \tag{3.24}$$

   This ensures the Y-axis is orthogonal to the X-axis. Normalize:

$$\hat{\mathbf{y}} = \frac{\mathbf{v}''_Y}{|\mathbf{v}''_Y|} \tag{3.25}$$

5. Z-axis (right-hand rule): the Z-axis is determined by the cross product:

$$\hat{\mathbf{z}} = \hat{\mathbf{x}} \times \hat{\mathbf{y}} \tag{3.26}$$

This ensures a right-handed coordinate frame.

The rotation matrix transforming from camera-centric coordinates to object-centric coordinates is:

$$\mathbf{R}_{\text{align}} = \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{y}} \\ \hat{\mathbf{z}} \end{bmatrix}^T \tag{3.27}$$

If user-specified IDs are not provided, the coordinate frame is derived automatically via principal component analysis (PCA). The covariance matrix of IDs positions is computed:

$$\mathbf{C} = \frac{1}{M} \sum_{i=1}^{M} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T \tag{3.28}$$

where $\bar{\mathbf{x}}$ is the centroid. The eigenvalue decomposition $\mathbf{C} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$ yields principal components. The axes are ordered by eigenvalue magnitude: the largest eigenvalue corresponds to the direction of maximum variance (X-axis), the second-largest to Y-axis, and the smallest to Z-axis. This automatic frame is centered at the marker centroid and aligned to the principal axes of the IDs distribution.

Once the target frame is defined by origin $\mathbf{o}$ and rotation $\mathbf{R}_{\text{align}}$, all 3D data are transformed:

$$\mathbf{x}_i^{\text{obj}} = \mathbf{R}_{\text{align}}(\mathbf{x}_i - \mathbf{o}) \tag{3.29}$$

The final output is a metric, object-centric 3D marker map ready for downstream pose estimation and tracking applications.

3.5.6 Rigid Transform from Marker to Probe Reference Frame Using 3D Slicer

In order to express the pose of the ultrasound scans in the camera reference frame, a rigid transform between the reconstructed marker map frame and the probe reference frame was required. This transform was obtained using 3D Slicer by registering to the same optical tracker frame both the 3D virtual marker map and the probe CAD model via stylus digitization. The marker to probe transform was then computed by composition.

The optical tracking system (NDI Polaris Vega) was connected to 3D Slicer via IGTLink. A stylus equipped with NDI passive markers provided the real-time transform to the tracker. The stylus tip location was estimated in the stylus reference frame using the pivot and spinning calibration.

After clamping the probe to prevent any motion during digitization, the stylus tip was placed on 20 physical corners (IDs) on the printed marker and the corresponding 3D points were collected in the tracker frame. A rigid transform was then applied to align the virtual marker map to the digitized points.

A CAD surface model of the probe was acquired using a 3D scanner and imported into 3D Slicer. The STL file was configured so that its embedded coordinate frame corresponds to the MFA reference frame described in (Lasso et al. 2014; PlusLib Developers n.d.), as illustrated in Fig. 3.6. Subsequently, a second fiducial registration was performed to align the STL model with the physical probe.

Figure 3.6: 3D model of the Clarius probe with the coordinate frame located on the transducer with MFA orientation.

Let ${}^{T}\mathbf{T}_{M}$ denote the homogeneous transform from the marker map frame M to the tracker frame T, and ${}^{T}\mathbf{T}_{P}$ the transform from the probe frame P to the tracker frame T. The desired marker-to-probe transform was obtained by:

$$ {}^{M}\mathbf{T}_{P} = \left({}^{T}\mathbf{T}_{M}\right)^{-1} {}^{T}\mathbf{T}_{P}. \tag{3.30} $$

3.6 6D Pose Estimation

Once the 3D marker map was established and aligned to the object coordinate frame, the tracking algorithm estimated the full six degree-of-freedom (6D) pose of the marker with respect to the camera in each incoming frame, i.e., its three-dimensional position and orientation. This section describes the pose estimation pipeline, which combined camera calibration, the perspective-n-point (PnP) problem, and robust outlier rejection via RANSAC.

3.6.1 Camera Calibration and Intrinsic Parameters

The camera intrinsic parameters and lens distortion characteristics had to be known to relate pixel coordinates to 3D rays in the camera coordinate frame. These parameters were typically estimated using a checkerboard calibration pattern and Zhang's method (Zhang 2000), implemented in standard computer vision libraries such as OpenCV's `calibrateCamera()` function.

3.6.1.1 Camera Matrix

The camera intrinsic parameters are encoded in the 3×3 camera matrix $\mathbf{K}$:

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3.31}$$

where f_x and f_y are the focal lengths in pixels, and c_x, c_y denote the principal point (image center). The matrix $\mathbf{K}$ projects homogeneous 3D points in the camera frame to homogeneous pixel coordinates:

$$\mathbf{p}_{\text{hom}} = \mathbf{K}\mathbf{X}_{\text{cam}} \tag{3.32}$$

where $\mathbf{X}_{\text{cam}}$ is a 3D point expressed in camera coordinates and $\mathbf{p}_{\text{hom}}$ is the homogeneous pixel coordinate.

3.6.1.2 Lens Distortion Correction

Real lenses exhibit radial and tangential distortion, causing pixels to deviate from the ideal pinhole projection model. The distortion is modeled via the Brown-Conrady model (Brown 1966):

$$x_{\text{distorted}} = x\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) + 2p_1 xy + p_2(r^2 + 2x^2) \tag{3.33}$$

$$y_{\text{distorted}} = y\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) + p_1(r^2 + 2y^2) + 2p_2 xy \tag{3.34}$$

where $r^2 = x^2 + y^2$ are normalized coordinates, k_1, k_2, k_3 are radial distortion coefficients, and p_1, p_2 are tangential distortion coefficients. During the calibration process, these coefficients are estimated and subsequently applied to convert distorted pixel coordinates to undistorted normalized image coordinates prior to PnP solving.

3.6.2 Perspective-n-Point Problem Formulation

The core task of pose estimation was to determine the marker's position and orientation with respect to the camera, given a set of correspondences between 3D points in the marker map and their 2D projections in the current image. This is formalized as the perspective-n-point (PnP) problem.

3.6.2.1 Problem Statement

Given a set of N 3D IDs positions $\{\mathbf{X}_i\}_{i=1}^N$ from the reconstructed map and their corresponding 2D image observations $\{\mathbf{z}_i\}_{i=1}^N$, the PnP problem sought the marker pose with respect to the camera (rotation $\mathbf{R}$ and translation $\mathbf{t}$) that best explains the observations:

$$\min_{\mathbf{R},\mathbf{t}} \sum_{i=1}^{N} \left\| \mathbf{z}_i - \text{proj}_K\left(\mathbf{R}\mathbf{X}_i + \mathbf{t}\right) \right\|^2 \tag{3.35}$$

where proj_K is the projection function incorporating the camera matrix $\mathbf{K}$ and lens distortion. The cost function represents the sum of squared reprojection errors: the discrepancy between observed pixel positions and the positions predicted by the pose hypothesis.

This nonlinear least-squares problem required a minimum of 6 constraints (3 for

rotation and 3 for translation) but is overdetermined when more than 6 correspondences are available, yielding improved accuracy through redundancy. Classical solvers included the Direct Linear Transform for initial estimates and iterative refinement via Levenberg-Marquardt optimization (Hartley et al. 2003).

The estimated pose $(\mathbf{R}, \mathbf{t})$ represented the rotation and translation vectors that express the 3D marker positions in the camera coordinate frame (OpenCV Documentation 2024b). A 3D point $\mathbf{X}_{\text{obj}}$ in tool/marker coordinates was transformed to camera coordinates via:

$$\mathbf{X}_{\text{cam}} = \mathbf{R}\mathbf{X}_{\text{obj}} + \mathbf{t} \tag{3.36}$$

3.6.3 RANSAC Outlier Rejection for Robust Estimation

In practice, the marker identification stage may yield incorrect correspondences due to noise, variable lighting conditions, or algorithmic instabilities. A robust estimation method is essential to identify and reject outliers while maintaining accuracy on inliers.

3.6.3.1 RANSAC Algorithm Overview

Random Sample Consensus (RANSAC) is a robust parameter estimation framework specifically designed to handle data contaminated by outliers (Fischler et al. 1981). The algorithm operated iteratively over a sequence of steps:

1. Sample: randomly select a minimal set of correspondences sufficient to uniquely determine the model (marker pose);
2. Fit: solve the PnP problem using only the sampled correspondences to obtain a candidate pose hypothesis;
3. Evaluate: project all 3D points using the candidate pose and compute reprojection errors. Count how many correspondences fall below a reprojection error threshold (inliers);
4. Select: retain the hypothesis with the largest inlier count after a fixed number of iterations (determined by the desired confidence level).

The number of RANSAC iterations was determined adaptively to ensure that, with high probability (e.g., 99.95%), at least one random sample is drawn from the inlier

set exclusively. When the inlier ratio is high, fewer iterations were required; conversely, when the inlier ratio is low, the algorithm requires more iterations to guarantee a good solution.

3.6.3.2 Application to PnP

In the pose estimation context, RANSAC operated on the 3D-2D correspondences derived from the marker map and current frame detections. The algorithm sampled minimal subsets, solved the PnP problem for each sample, and evaluated each candidate pose by computing reprojection errors and counting inliers below a specified threshold. The final pose estimate was the hypothesis that maximized the inlier count.

3.6.3.3 Warm-Starting with Prior Pose

To accelerate convergence and reduce the risk of converging to suboptimal solutions, the pose estimation process employed warm-starting: the marker pose estimated from the previous frame was supplied as an initial guess to the RANSAC solver. This prior pose was used to initialize the iterative refinement step (via Levenberg-Marquardt). This strategy was essential in video sequences where camera motion is continuous and small between successive frames. In such cases, the prior pose was typically close to the true current pose, enabling the optimizer to converge rapidly without requiring extensive exploration of the parameter space. Without warm-starting, the solver could have required substantially more iterations or converged to local minima, particularly in scenarios marker identification becomes ambiguous.

3.6.3.4 Post-RANSAC Validation Filters

After RANSAC completes and produces a pose estimate, two kinematic filters were applied to detect and reject physically implausible results:

- Motion Threshold: the translational distance between the current and previous frame poses was computed. If this distance exceeds an adaptive threshold (calibrated to the expected object speed and operating distance range), the pose was rejected as a tracking failure;

- Rotation Magnitude Check: the rotation vector (angle-axis representation) is converted to a rotation angle. If the angle magnitude differed excessively from the previous frame (indicating an unrealistic instantaneous rotation), the pose was rejected.

These filters prevented the tracker from temporarily adopting infeasible configurations resulting from ambiguous detections or false correspondences. When a pose was rejected, the tracker reset state and re-initialized on the next frame containing a sufficient number of detections.

3.6.4 Pose Refinement via Unidentified Corners (Preliminary Implementation)

The marker detection stage yielded two sets of corners: identified and unidentified. Identified corners were those for which the detector assigned an ID with sufficient confidence. Unidentified corners were detected ones whose ID assignment was uncertain and therefore withheld. This conservative choice was adopted to avoid propagating potentially incorrect correspondences. Although unidentified corners are excluded from the primary RANSAC estimation, they still contained useful geometric information and could improve pose accuracy if they can be reliably matched to map points.

A secondary pose refinement stage (currently implemented but not yet validated experimentally) attempted to leverage unidentified corners by matching them to the 3D marker map:

1. Projection: using the marker pose obtained from RANSAC (Stage 1), project all 3D marker positions onto the image plane via the projection function;

2. Geometric Matching: for each unidentified corner, compute the distance to the nearest projected marker point. Accept the match if the distance is below a threshold;

3. Augmented PnP: combine the originally identified inlier set with the newly matched unidentified corners to form an augmented set of 3D-2D correspondences. Re-solve the PnP problem using this enlarged set, initializing from the Stage 1 pose.

3.6.5 Localization of the Ultrasound Image Plane in the Camera Frame

Once the constant rigid transform from the marker map frame to the ultrasound image plane, ${}^{M}\mathbf{T}_{P}$, had been computed in 3D Slicer (as described in Section 3.5.6), it was combined with the per-frame pose estimated by the tracker. In each video frame, the PnP+RANSAC stage provided the marker map pose with respect to the camera, ${}^{C}\mathbf{T}_{M}$. The raw pose parameters were first smoothed using a nonuniform Savitzky-Golay filter to reduce noise and handle temporal irregularities in the marker detections (Gorry 1991). The smoothed poses were then interpolated to align with the Clarius ultrasound acquisition timestamps. Finally, the probe pose in the camera frame was obtained by composition of the two rigid transformations:

$$ {}^{C}\mathbf{T}_{P} = {}^{C}\mathbf{T}_{M}(t)\,{}^{M}\mathbf{T}_{P}. \tag{3.37} $$

This formulation yielded, for each acquired ultrasound frame, the spatial location and orientation of the corresponding image plane with respect to the camera coordinate system. A schematic summary of the reference frames and transformations involved was shown in Figure 3.7.

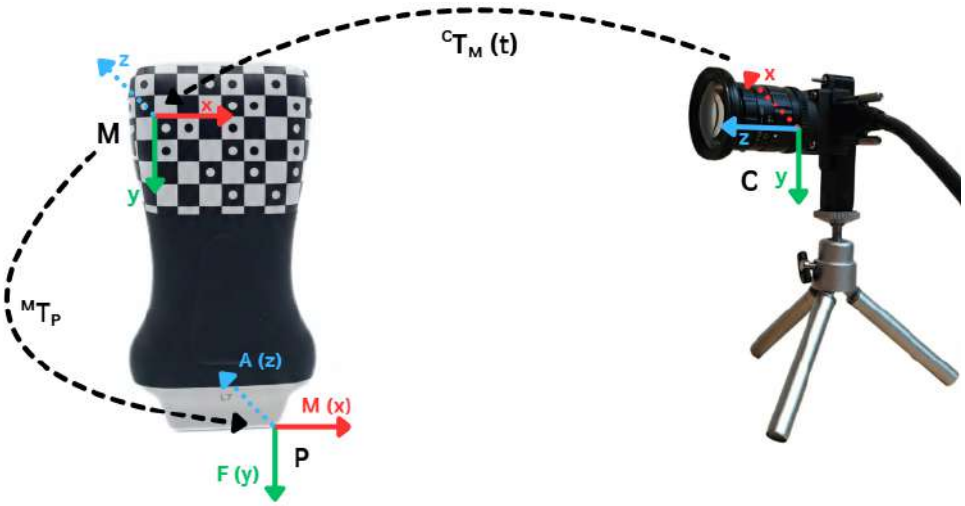


Figure 3.7: Definition of the reference frames of the system. The probe frame $\{P\}$, the marker frame $\{M\}$, and the camera frame $\{C\}$ are shown together with their respective axes. The rigid transformation is given by ${}^{C}\mathbf{T}_{P} = {}^{C}\mathbf{T}_{M}(t)\,{}^{M}\mathbf{T}_{P}$, where ${}^{C}\mathbf{T}_{M}(t)$ is obtained from marker tracking.

A complete operator-oriented, step-by-step description of the full pipeline (from

marker design/printing to calibration, acquisition, and Slicer-based transform estimation) is provided in Appendix B.

Chapter 4

Methods: Validation of Tracking Algorithm

4.1 Experimental Setup and Evaluation Protocol

4.1.1 Ground Truth Reference Systems

To quantitatively validate the tracking system and characterize its pose estimation accuracy, two precision mechanical stages were employed to generate ground truth motion trajectories. For this application, the target performance was to match the sub-millimetric translational accuracy and sub-degree rotational accuracy previously achieved with the Inside-Out tracking architecture. These stages provided controlled, repeatable transformations with well characterized accuracy specifications, enabling rigorous measurement of tracking performance across a range of motion magnitudes and directions.

4.1.1.1 Rotational Stage

The rotational stage used in validation experiments was a Newport M-UTR80 precision manual rotation stage (Micro Contrôle, France), manufactured by Newport Corporation and illustrated in Fig. 4.1. This stage provided fine-tuning angular orientation with two-axis control: coarse rotation over 360° with free rotation via a side-mounted clutch, and fine adjustment over a $\pm 2^\circ$ range for precise positioning. The stage was constructed from hardened tool steel with preloaded precision bearings, ensuring low eccentricity and smooth motion. The angular resolution was $1\,\mathrm{arcminute}$ $(1/60\,^\circ)$ via vernier and scale

markings, and the maximum angular repeatability was $\pm 60 \times 10^{-6}\,\mathrm{rad}$ ($\approx 0.01°$). No explicit specification of absolute angular accuracy was provided. However, the combination of fine resolution and high repeatability indicated that systematic stage errors were negligible compared to the sub-degree accuracy target used to benchmark the tracking system.

The specified angular resolution and repeatability were nominal catalogue specifications provided by the manufacturer; no individual calibration certificate was available for the specific stage used in this study.

The M-UTR80 provided deterministic angular positioning with sub-degree precision, making it suitable for validating the tracker's sensitivity to rotational perturbations. The vernier scale markings enabled angular displacements to be set and verified with high repeatability.

Figure 4.1: Rotational stage (Newport M-UTR80) used for ground truth angular displacements.

4.1.1.2 Translational Stage

The translational stage was a Velmex UniSlide® A25 Series manual linear actuator (model A250601-S2.5, Velmex Inc., USA), shown in Fig. 4.2. This compact stage provided precise linear motion along a single axis via a user-operated lead screw drive mechanism. The A25 Series was constructed from aluminum extrusions with low-friction polymer bearing pads and was widely used in precision measurement and alignment applications. The instrument featured a $1\,\mathrm{mm}$-per-revolution lead screw with graduated knob markings, enabling direct position readout with approximately $0.01\,\mathrm{mm}$ resolution per knob division. The stage exhibited repeatability of $\pm 0.025\,\mathrm{mm}$ and straight-line accuracy of $0.18\,\mathrm{mm}$

per $25.4\,\mathrm{cm}$ of travel with the standard lead screw configuration, corresponding to approximately $\pm 0.07\,\mathrm{mm}$ over the $88.9\,\mathrm{mm}$ travel distance. These values were nominal performance specifications reported in the manufacturer's datasheet rather than results of a traceable factory calibration of the specific unit used.

The A25 UniSlide provided manual displacement control that was sufficiently repeatable and accurate to impose known translations at the experimental working distances.

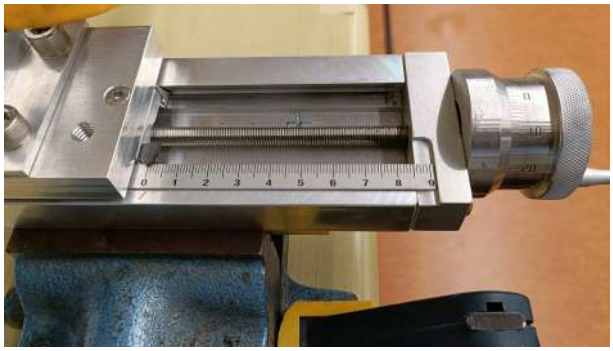


Figure 4.2: Translational stage (Velmex UniSlide® A25 Series) used for ground truth linear displacements.

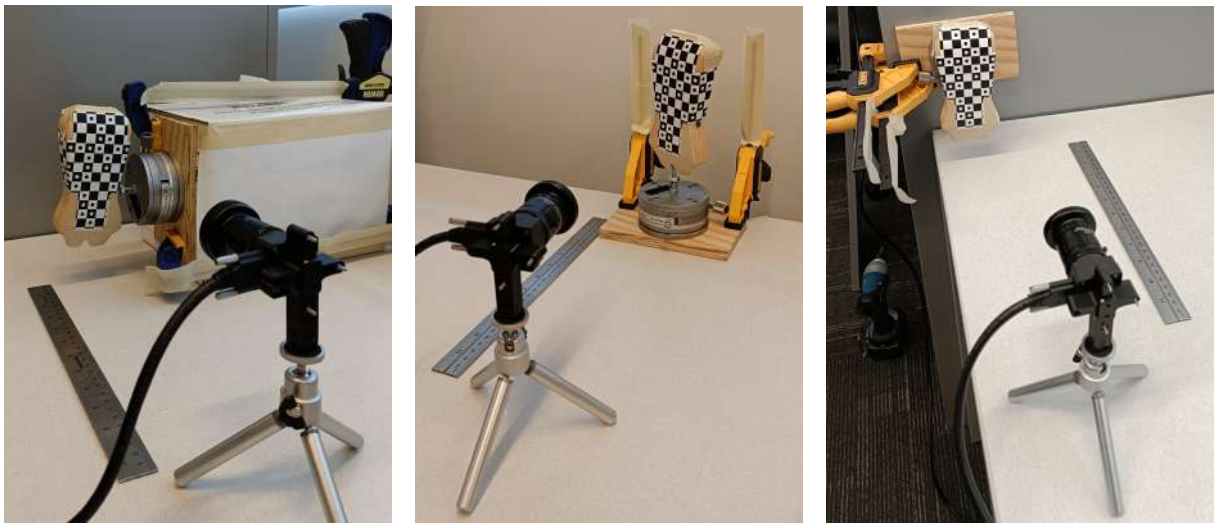
4.1.2 Test Protocols

4.1.2.1 Camera Calibration

Calibration was first performed to determine the camera intrinsic parameters and lens distortion coefficients using a ChArUco board with 7×9 squares and 5×5 ArUco markers positioned at the required working distance as illustrated in Fig. 4.3. The calibration was considered satisfactory with a reprojection error below 0.2 px.

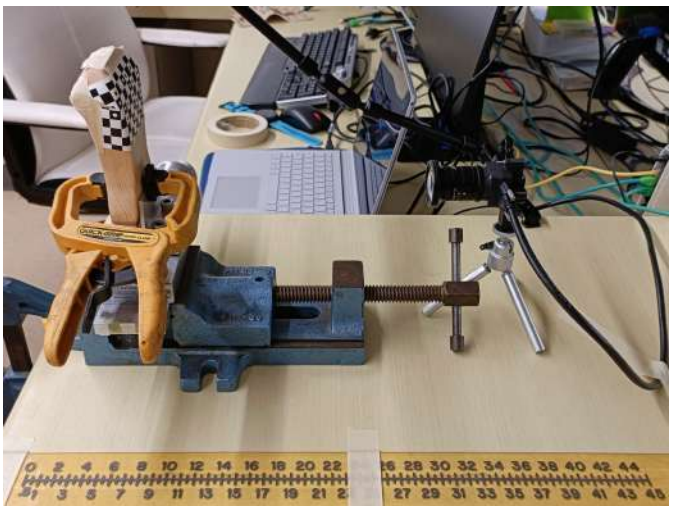
All validation experiments were conducted with the camera mounted orthogonally to the precision stages at a fixed working distance of $300\,\mathrm{mm}$ and a recording frequency of $20\,\mathrm{Hz}$, to limit post-processing time. A wooden probe mock-up equipped with the printed marker was then rotated and translated about/along the transverse, beam, and elevation axes, as shown in Fig. 4.4 for the rotational stage and Fig. 4.5 for the translational stage.

Figure 4.3: Calibration setup showing the ChArUco board positioned at the working distance for intrinsic parameter and distortion coefficient estimation.

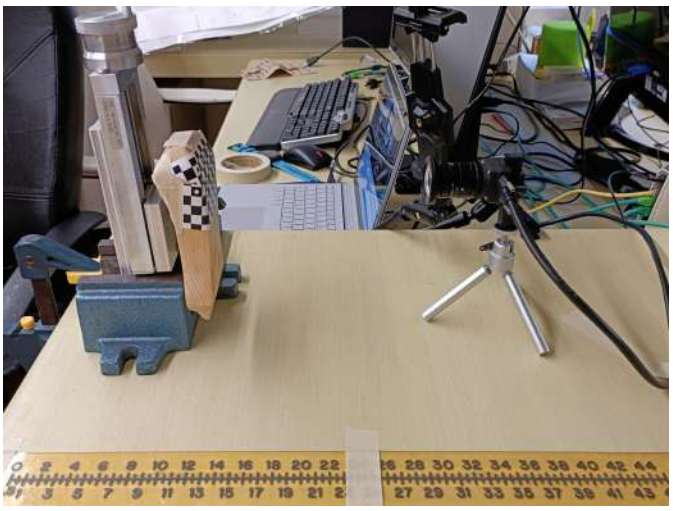


Transverse Axis Beam Axis Elevation Axis

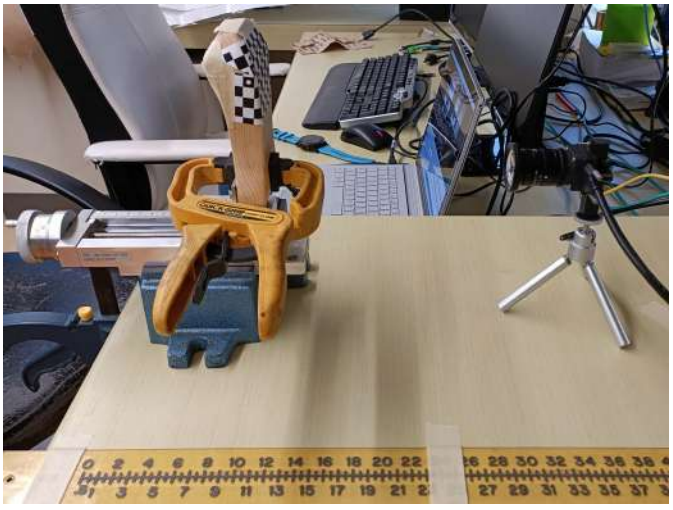
Figure 4.4: Rotation test configuration for the 12×13 HydraMarker. The probe prototype was rotated about the three axes using the rotational stage.



Transverse Axis



Beam Axis



Elevation Axis

Figure 4.5: Translation test configuration for the 6×10 HydraMarker. The probe prototype was displaced along the three axes using the translational stage.

4.1.2.2 Static Test

A preliminary static test was performed to verify correct marker detection and identification after fabrication. A small number of frames was acquired at different marker orientations, and the full detection pipeline was executed to confirm that the marker corners were consistently localized and correctly labeled.

4.1.2.3 Repeatability Testing

Repeatability was evaluated by repeatedly returning the stage to a reference position and applying the same displacement. Stage readings were documented before and after each motion using external photographs of the stage scale. For each trial, the system was held stationary for approximately $2\,\mathrm{s}$ both before and after the motion so that the corresponding ~ 40 frames were averaged to reduce measurement jitter (random error). The repeatability metric was computed as the difference between the measured poses at the initial and final holds.

For rotational repeatability, two acquisition sequences were performed: 30 repetitions from 0° to 45°, followed by 30 repetitions from 45° to -45°.

For translational repeatability, the displacement magnitude depended on the axis and on the specific experimental campaign; the corresponding step sizes were therefore reported together with the results for each experiment.

The experimental protocol followed an iterative approach: an initial design was tested, selected variables were modified to improve performance, and targeted tests were repeated to verify the effect of each modification. Once the final design was selected, the full set of experiments was repeated with a reduced number of repetitions, under the assumption that prior results remained valid.

4.1.2.4 Accuracy Testing Over Incremental Displacements

Accuracy was assessed by applying known incremental motions starting from a central reference position. The stage was rotated or translated by fixed increments of 5° or $10\,\mathrm{mm}$ and held for approximately $2\,\mathrm{s}$ at each step to enable frame averaging. These step sizes were chosen to be representative of clinically relevant probe repositioning during Graf hip examinations, i.e., angular and translational adjustments that can realistically occur between repeated sweeps on the infant pelvis. For each increment, the displacement

relative to the reference position was computed. The procedure was then repeated in the opposite direction.

4.2 Evaluation Metrics

All experiments were conducted under normal, uncontrolled indoor lighting conditions without supplementary illumination. This choice was intentional to approximate typical clinical room illumination rather than highly controlled photometric conditions, so that the reported tracking performance reflects realistically achievable lighting scenarios.

4.2.1 Accuracy Assessment

4.2.1.1 Jitter Analysis

Jitter analysis quantifies the temporal stability and noise characteristics of the estimated pose in a stationary configuration. In this assessment, the camera and marker remain at fixed positions while pose estimates are continuously computed across successive video frames. Any frame to frame variations in the estimated pose represent tracking noise and algorithmic uncertainty, independent of any true motion. By measuring this intrinsic variability, jitter analysis provides a direct quantification of the lower bound on pose estimation accuracy and the signal-to-noise ratio achievable by the tracking system.

The jitter analysis was performed on a sequence of 2005 consecutive video frames captured over 100 seconds during a stationary measurement trial. For each frame, the pose estimation pipeline produced a rotation matrix $\mathbf{R}$ and translation vector $\mathbf{t}$, representing the marker pose with respect to the camera. Pose estimates across frames were then analyzed separately for translational and rotational components.

The rotational component is quantified using the angle-axis representation on $\mathrm{SO}(3)$. For each frame, the deviation from the mean rotation is computed as:

$$\theta_i = \left\| \Delta\mathbf{R}_i^{\text{axis}} \right\| \tag{4.1}$$

where $\Delta\mathbf{R}_i = \mathbf{R}_i \mathbf{R}_{\text{mean}}^{-1}$ represents the relative rotation between the frame's estimated rotation and the mean rotation, and $\Delta\mathbf{R}_i^{\text{axis}}$ denotes the angle-axis representation of this relative rotation. The norm of the angle-axis vector directly yields the rotation angle

in radians. The sample standard deviation of these angular deviations provides the rotational jitter magnitude.

The translational component is quantified as the Euclidean distance from each frame's estimated position to the mean position computed across all frames:

$$d_i = \|\mathbf{t}_i - \bar{\mathbf{t}}\|_2 \tag{4.2}$$

where $\mathbf{t}_i$ is the translation vector for frame i and $\bar{\mathbf{t}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{t}_i$ is the mean position. The sample standard deviation of these distances provides the translational jitter magnitude.

Both metrics are computed with 95% confidence intervals derived from chi-squared distributed statistics on the sample variance. Specifically, the confidence interval bounds are computed via:

$$\sigma_{\text{CI, lower}} = \sqrt{\frac{(n-1)s^2}{\chi^2_{1-\alpha/2}(n-1)}}, \quad \sigma_{\text{CI, upper}} = \sqrt{\frac{(n-1)s^2}{\chi^2_{\alpha/2}(n-1)}} \tag{4.3}$$

where s^2 is the sample variance, $n = 2005$ is the number of samples, $\alpha = 0.05$, and χ^2 denotes the chi-squared critical value.

The jitter analysis results from the stationary trial are presented in Table 4.1. These results represent the intrinsic accuracy limit of the tracking system independent of any algorithmic or calibration uncertainties. The jitter metrics serve as an upper bound on the measurement precision achievable by the marker-based tracker in ideal, stationary conditions.

Table 4.1: Jitter analysis results quantifying temporal pose stability in a stationary configuration.

Metric	**Sample Std Dev**	**95% Confidence Interval**
Rotational Jitter [°]	0.010	[0.0099, 0.0105]
Translational Jitter [mm]	0.005	[0.0052, 0.0056]

4.2.1.2 Rotational Error

Rotational error quantifies the discrepancy between HydraMarker based pose estimates and ground truth rotations measured via caliper across multiple trials. For each trial, the rotation change is computed by averaging the first and last $N = 20$ valid frames and calculating the relative rotation between these averaged poses.

For the averaged start rotation $\mathbf{R}_{\text{start}}$ and end rotation $\mathbf{R}_{\text{end}}$, the rotation delta is computed as:

$$\Delta\mathbf{R} = \mathbf{R}_{\text{end}}\mathbf{R}_{\text{start}}^{-1} \tag{4.4}$$

The rotation angle error is extracted from the angle-axis representation:

$$\theta_{\text{error}} = \left\|\Delta\mathbf{R}^{\text{axis}}\right\| \tag{4.5}$$

For each trial, the tracked delta angle is compared to the ground truth caliper measurement, and the per-trial error is:

$$e_i = \theta_{\text{tracked},i} - \theta_{\text{caliper},i} \tag{4.6}$$

Mean error and confidence intervals are then computed across all trials.

4.2.1.3 Translational Error

Translational error quantifies the magnitude of position estimation discrepancy between the HydraMarker tracker and caliper-based ground truth measurements. For each trial, the translation change is computed by averaging the first and last $N = 20$ valid translation vectors and calculating the Euclidean distance between these averaged positions.

The start and end mean positions are:

$$\mathbf{t}_{\text{start}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{t}_i, \quad \mathbf{t}_{\text{end}} = \frac{1}{N}\sum_{i=n-N+1}^{n}\mathbf{t}_i \tag{4.7}$$

The translation delta is the Euclidean distance:

$$\Delta t = \left\|\mathbf{t}_{\text{end}} - \mathbf{t}_{\text{start}}\right\|_2 \tag{4.8}$$

Per-trial error is computed as:

$$e_i = \Delta t_{\text{tracked},i} - \Delta t_{\text{caliper},i} \tag{4.9}$$

Mean error and confidence intervals are then computed across all trials.

4.2.2 Robustness Assessment

4.2.2.1 Single Marker Occlusion Tolerance Test

The first test assessed occlusion robustness using a 12×12 HydraMarker with 4×4 ID tags mounted on a curved surface. To evaluate identification performance under progressive occlusion, a sequence of images was captured with the marker progressively hidden until the identification algorithm failed to correctly decode the ID. This qualitative test served as an exploratory assessment to understand occlusion limits before finalizing the probe marker design.

4.2.2.2 Dual-Marker Detection and Identification Test

The second test involved simultaneous detection and identification of two distinct markers within a single camera frame, each affixed to a curved surface. Both markers were 10×10 fields with ID tag dimensions of 3×3 and 2×5 respectively. To ensure the two marker fields were distinct, both were generated using the same HydraMarker generator but with 30% of cells randomly shuffled and locked, guaranteeing different marker fields.

4.2.2.3 Multi-Marker Identification Algorithm

When multiple markers are present in the camera frame, the identification algorithm must correctly assign each detected feature grid to the appropriate marker field. Initially, the algorithm employed a greedy approach: for each detected Delaunay triangulation grid in the image, it sequentially attempted to match against all available marker field matrices (sta matrices). Specifically, for each grid, the algorithm performed $m \times n$ matching operations, where m is the number of marker field matrices and n is the number of computed dot tables (angle-based feature descriptors). Each matching operation tested all four rotations of the grid against the marker field, resulting in significant computational redundancy: identical corner detection and feature descriptor computation were repeated for each marker field. Although this approach yielded results, it was computationally inefficient and produced multiple candidate matches per grid (4 matches per grid with 2 marker fields), requiring post-hoc disambiguation.

To improve efficiency and matching reliability, the algorithm was restructured to compute feature descriptors once and cache them for reuse across all marker fields. In the

optimized approach: (1) dot tables, point grids, and dot-center lists derived from each Delaunay triangulation are computed a single time and stored; (2) each detected grid is then matched against all candidate marker field matrices using the cached feature data, eliminating redundant corner detection; (3) for each grid-to-marker-field pairing, all four rotations are tested, and a match score is recorded along with coverage (number of aligned cells), rotation, and decoded ID array; (4) the grid is then assigned to the marker field yielding the highest match score. This approach decouples feature extraction from marker field matching, reducing computational cost while improving the reliability of multi-marker discrimination by enabling direct score-based comparison between candidate marker fields for each detected region.

Chapter 5

Results

Outliers exceeding 3σ were identified in multiple trials, prompting review of the corresponding image sequences to determine the cause. In total, 11 outliers were detected over 666 acquired samples (approximately 1.7% of all measurements). In these cases, the probe was either inadvertently constrained by contact with the operator's hand during stage actuation, preventing a full motion, or the acquisition system dropped frames due to throughput limitations. The affected samples were therefore excluded from the dataset. For each experiment, the number of valid trials is summarized in the corresponding table.

5.1 Rotational Error

5.1.1 Design 1: 12×13 Balanced HydraMarker

Rotational accuracy was first evaluated using the cross-shaped 12×13 HydraMarker (3×3, 5×5, and 1×13 ID tags) illustrated in Fig. 2.9 (b), corresponding to the balanced solution selected from the design optimization. For each axis, 30 repeated trials were collected in two angular ranges ($0°$– $45°$ and $45°$– $315°$, passing through $0°$) using the experimental configuration shown in Fig. 4.4. A virtual 3D map of the marker was reconstructed following the procedure in Section 3.5 (Fig. 5.1); the reprojection error after bundle adjustment (BA) was 0.112 px. Rotation tracking errors are reported in Fig. 5.2, while Fig. 5.3 shows the corresponding mean-subtracted distributions, highlighting repeatability within each axis/range subset. Table 5.4 summarizes per-configuration error statistics for these trials.

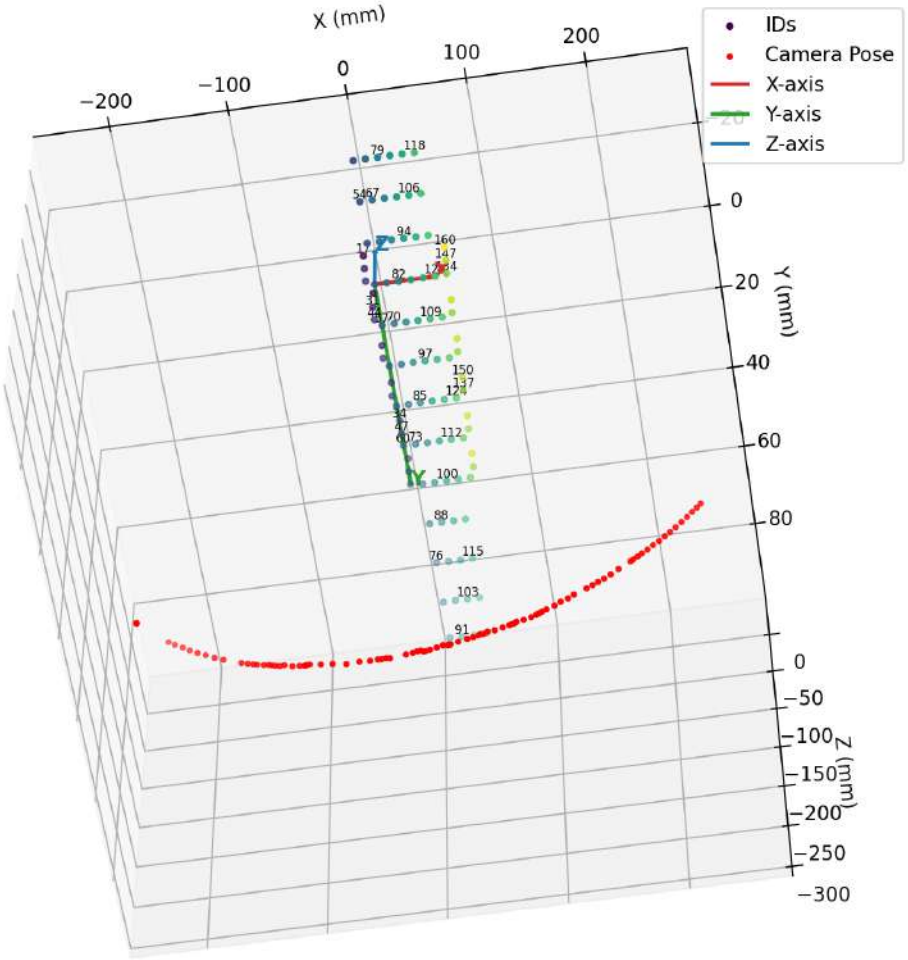


Figure 5.1: Structure from Motion reconstruction of the 12×13 HydraMarker used for pose estimation. Marker corner points are shown with a viridis colormap and estimated camera poses are shown in red. The marker reference frame is indicated by the axes (X red, Y green, Z blue).

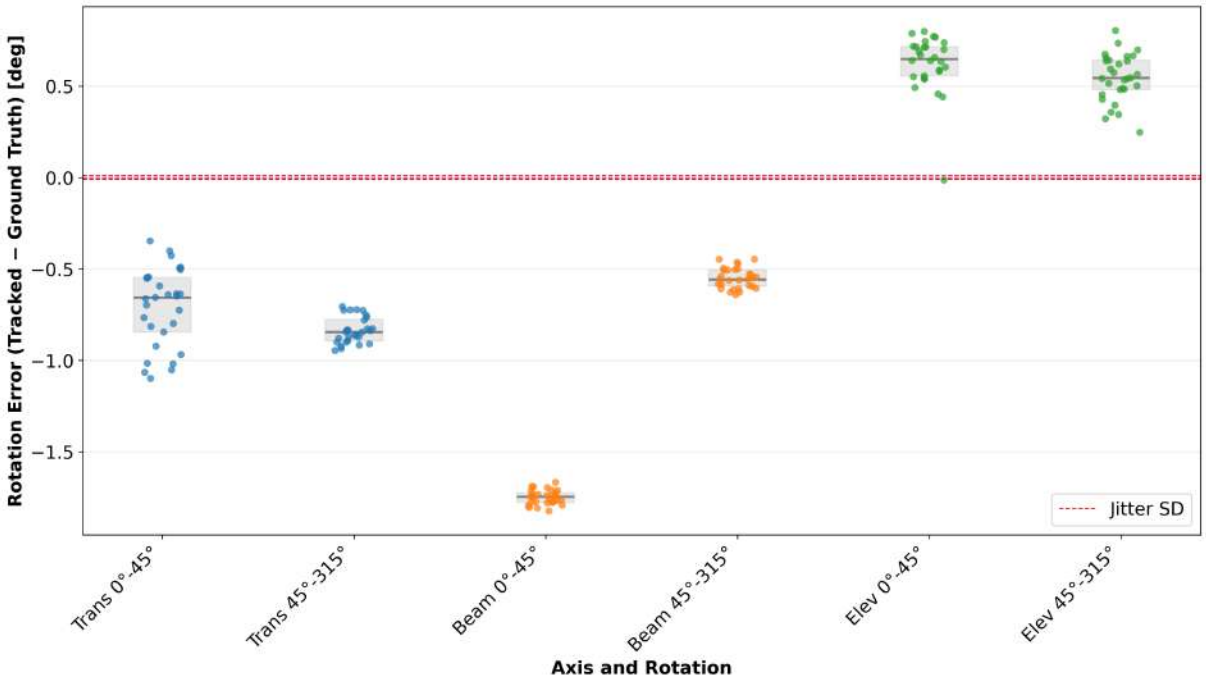


Figure 5.2: Rotation tracking error for the three axes (transverse in blue, beam in orange, elevation in green) evaluated over two rotation ranges ($0°$– $45°$ and $45°$– $315°$). Each point is an individual trial and gray median/IQR overlays summarize each configuration. Red dashed lines indicate the rotational jitter standard deviation reference level ($0.010°$).

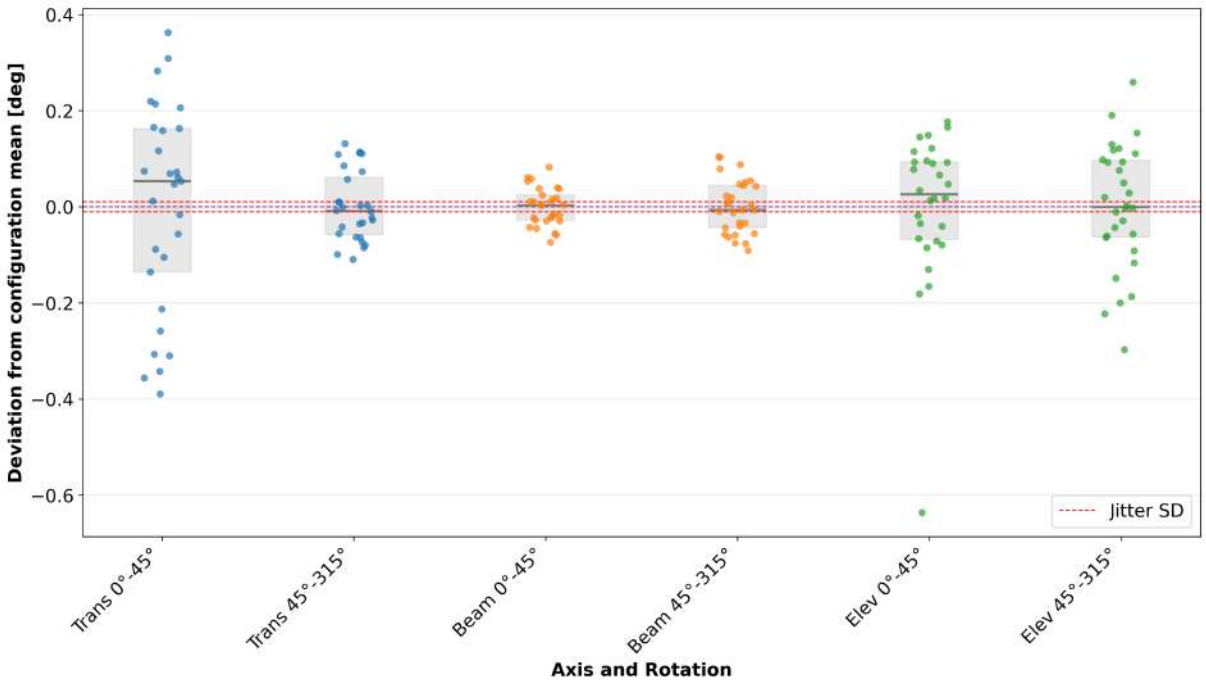


Figure 5.3: Mean-subtracted rotation tracking error for Fig. 5.2, obtained by subtracting the mean error within each axis/range subset. Points show trial-wise deviations from the configuration mean and gray median/IQR overlays summarize spread. Red dashed lines indicate the rotational jitter SD reference level from the static test ($0.010°$).

5.1.2 Design 2: 15×17 Maximum-Corners HydraMarker

The initial results indicated comparatively larger errors during rotation about the beam axis (Fig. 5.2). To investigate whether increased feature density could improve pose conditioning, additional experiments were performed using the maximum corners design. The marker was modified to a 15×17 field with 4×4 and 2×8 ID tags (Fig. 5.4), while maintaining the same overall dimensions. The ID tag layout was also adjusted to better represent the probe side surfaces. Testing was restricted to the beam axis over $0°$– $45°$. The reconstructed virtual map (Fig. 5.5) achieved a reprojection error of 0.128 px (after BA), and the resulting tracking error and mean-subtracted repeatability are shown in Figs. 5.6 and 5.7, with Table 5.5 reporting per-configuration summary metrics. In this configuration, rotation about the beam axis exhibited an approximately $0.6°$ systematic bias relative to the stage reference, as visible in Fig. 5.6.

Figure 5.4: 15×17 HydraMarker with 4×4, and 2×8 ID tags mounted on the probe prototype for rotation testing around the beam axis.

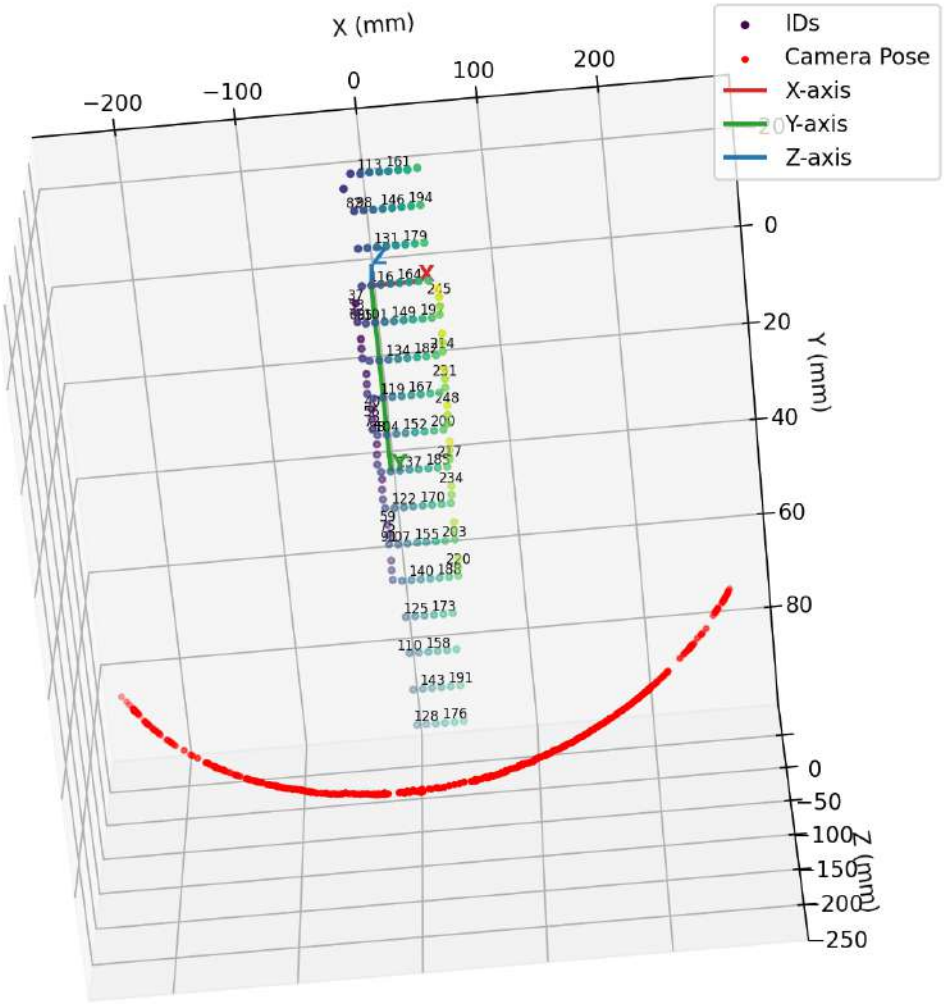


Figure 5.5: Structure from Motion reconstruction of the 15×17 HydraMarker used for pose estimation. Marker corner points are shown with a viridis colormap and estimated camera poses are shown in red. The marker reference frame is indicated by the axes (X red, Y green, Z blue).

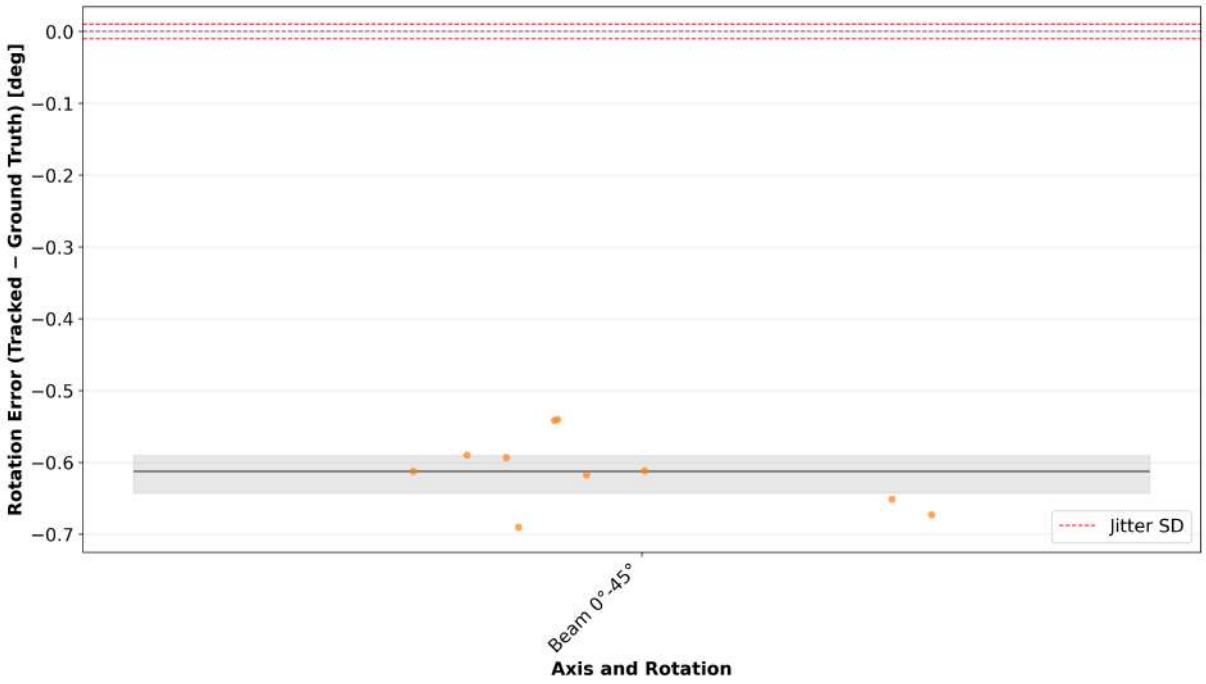


Figure 5.6: Rotation tracking error for the beam axis over the $0°$– $45°$ range. Each point is an individual trial and gray median/IQR overlays summarize each configuration. Red dashed lines indicate the rotational jitter standard deviation reference level ($0.010°$).

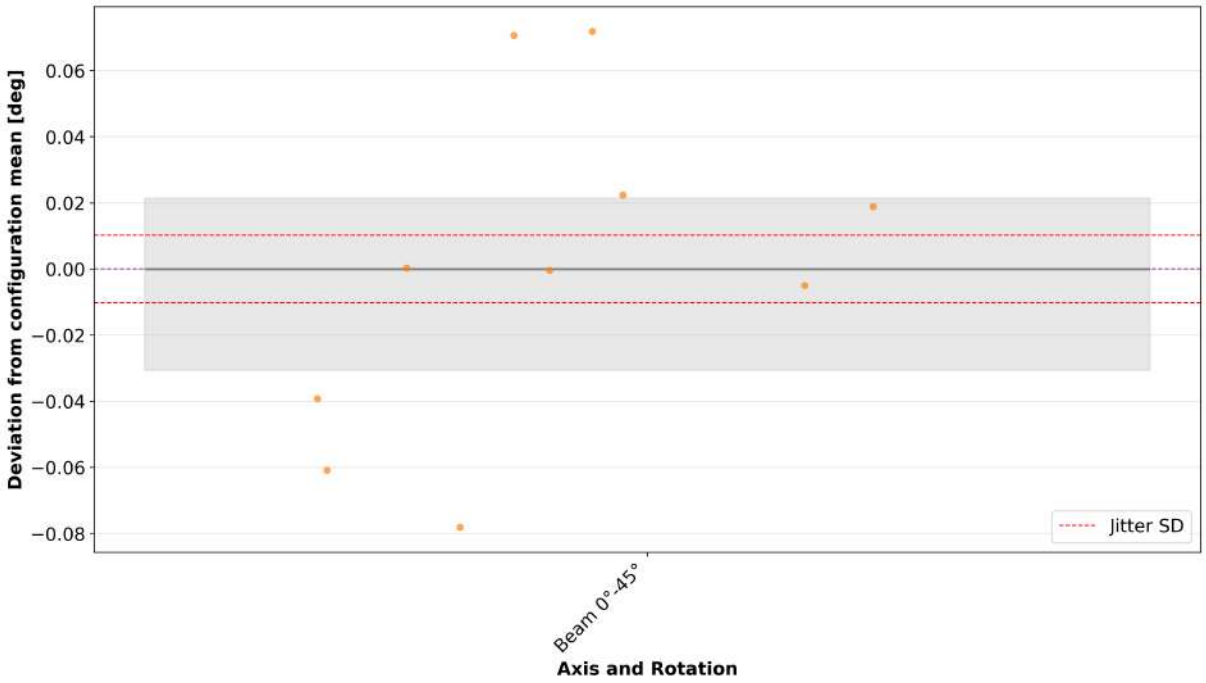


Figure 5.7: Mean-subtracted rotation tracking error for Fig. 5.6, obtained by subtracting the mean error. Points show trial-wise deviations from the configuration mean and gray median/IQR overlays summarize spread. Red dashed lines indicate the rotationaljitter SD reference level from the static test ($0.010°$).

5.1.3 Design 3: 6×10 Final HydraMarker

Although the 15×17 configuration improved both accuracy and repeatability, the marker design was further refined to better match clinical handling and to eliminate regions expected to be permanently occluded by the sonographer's hand. Figure 5.8 (a) illustrates the representative grip, and Fig. 5.8 (b) shows the resulting rectangular 6×10 HydraMarker (3×3, 1×10, and 6×2 ID tags) positioned to remain visible under this grip. The dimensions of the marker are $48\,\mathrm{mm} \times 80\,\mathrm{mm}$. Given the substantial redesign, the full three axis protocol was repeated with 10 trials per rotation. The corresponding virtual map is shown in Fig. 5.9, with a reprojection error of 0.081 px (after BA). The final rotational error distributions and mean-subtracted repeatability results are reported in Figs. 5.10 and 5.11, with Table 5.6 summarizing the corresponding configuration-level statistics.

(a) (b)

Figure 5.8: (a) Representative clinical grip of the ultrasound probe during scanning. (b) Probe prototype instrumented with a 6×10 HydraMarker that would be visible when held as shown in (a).

Further accuracy testing was performed with the final 6×10 design to characterize drift over a full incremental rotation sequence. Starting from a perpendicular marker

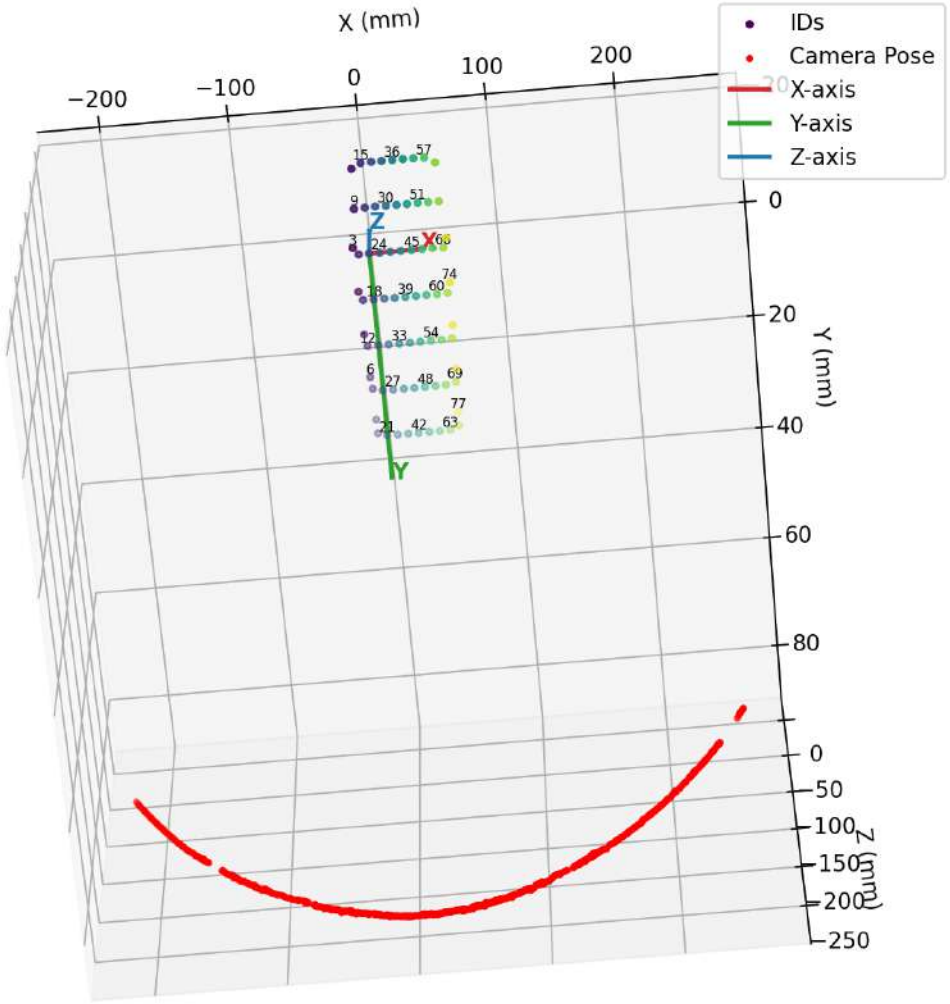


Figure 5.9: Structure from Motion reconstruction of the 6×10 HydraMarker used for pose estimation. Marker corner points are shown with a viridis colormap and estimated camera poses are shown in red. The marker reference frame is indicated by the axes (X red, Y green, Z blue).

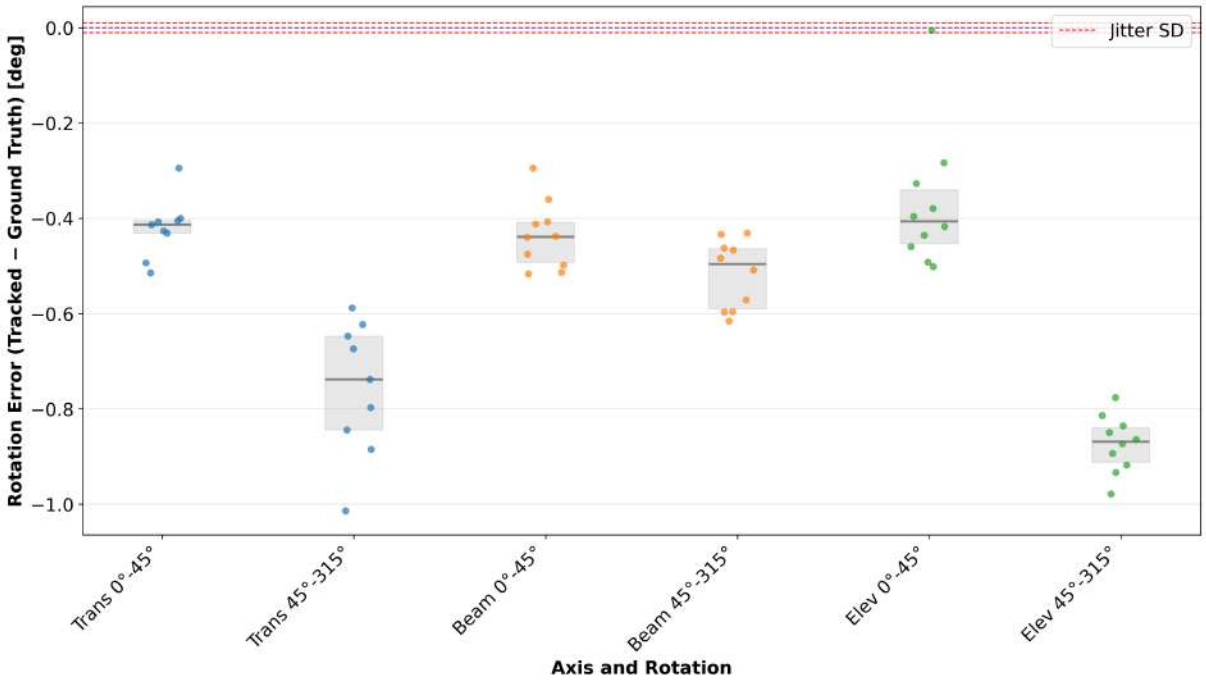


Figure 5.10: Rotation tracking error for the three axes (transverse in blue, beam in orange, elevation in green) evaluated over two rotation ranges ($0°$– $45°$ and $45°$– $315°$). Each point is an individual trial and gray median/IQR overlays summarize each configuration. Red dashed lines indicate the rotational jitter standard deviation reference level ($0.010°$).

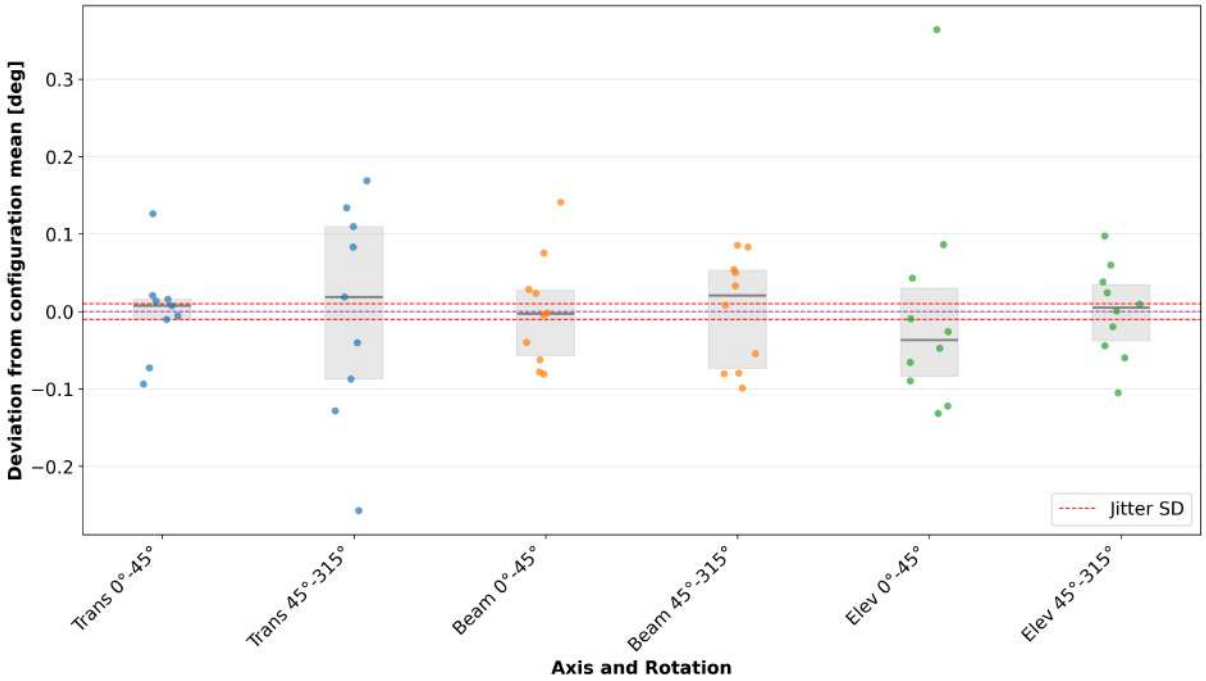


Figure 5.11: Mean-subtracted rotation tracking error for Fig. 5.10, obtained by subtracting the mean error within each axis/range subset. Points show trial-wise deviations from the configuration mean and gray median/IQR overlays summarize spread. Red dashed lines indicate the rotational jitter SD reference level from the static test ($0.010°$).

orientation relative to the camera, two trials were executed per axis: an increment trial in which the stage was advanced from 0° in 5° steps following the right-hand rule, and a decrement trial that repeated the sequence from 0° in the opposite direction. The accuracy plots in Figs. 5.12 – 5.16 show the tracked rotation error relative to the rotational stage angle for both increment and decrement trials, and each plot includes a slope-only least-squares fit to capture systematic drift across steps. The accompanying tables (Tables 5.1– 5.3) report mean error, variability, and aggregate error metrics for each direction and for the combined data. Representative error profiles for transverse, beam, and elevation rotations are shown in Figs. 5.13–5.17, where each point denotes the mean difference between tracked and caliper rotation at a given step and error bars indicate jitter variability.

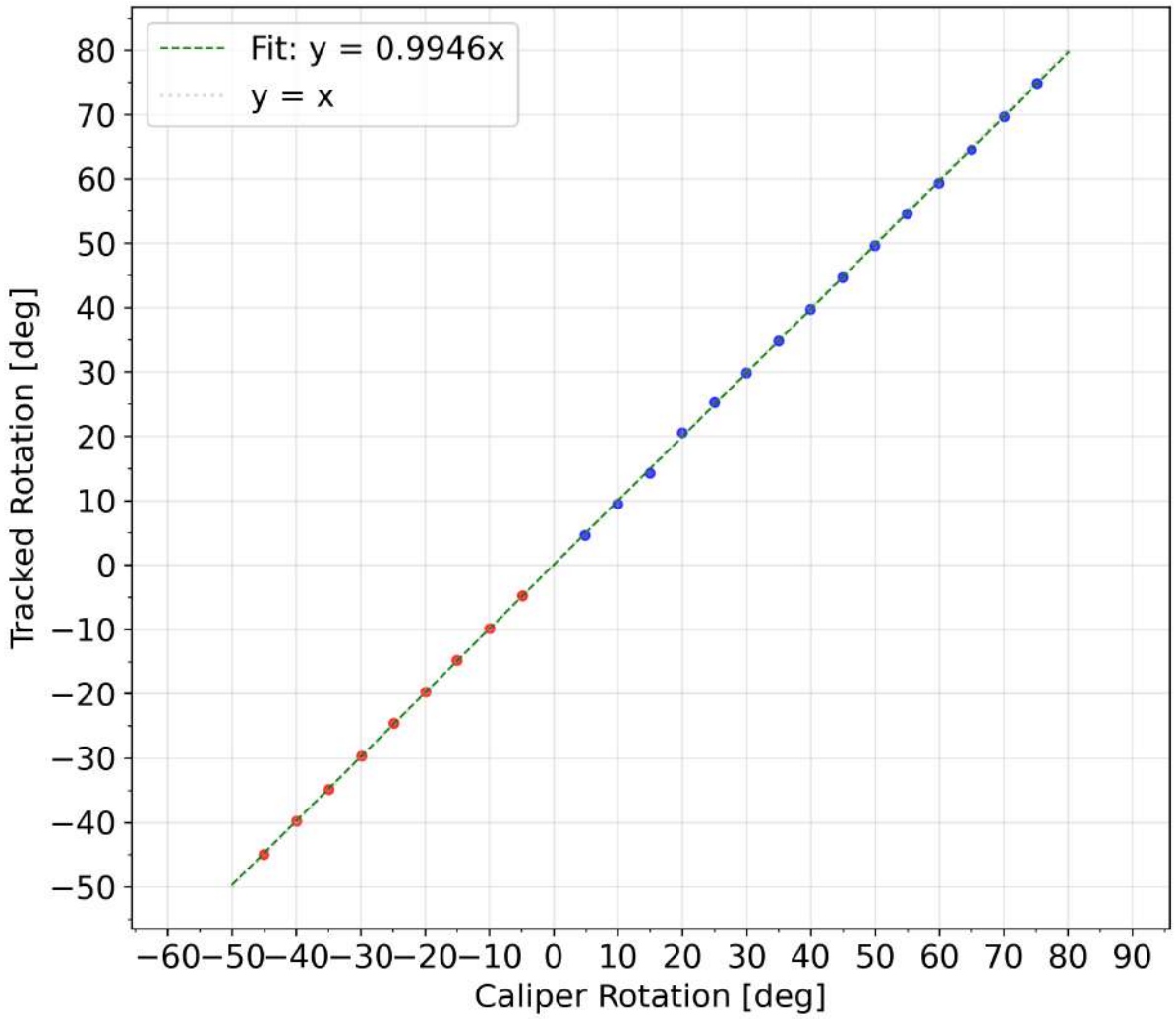


Figure 5.12: Rotation accuracy about the transverse axis, with increment (blue) and decrement (red) measurements relative to the caliper reference with the identity relation (gray) and best-fit regression (green).

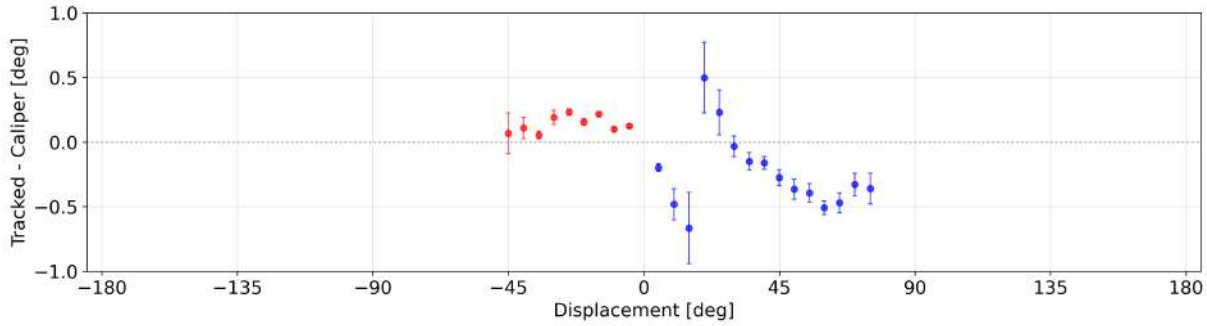


Figure 5.13: Transverse rotation error as a function of angular displacement. The points show the mean difference between the algorithm estimate and the caliper reading for each step, and the error bars represent the standard deviation of the fluctuations measured while the system is held still.

Table 5.1: Summary statistics for rotation accuracy about the transverse axis, corresponding to Fig. 5.12.

Condition	N	Mean [deg]	CI low [deg]	CI high [deg]	RMSE [deg]	MAE [deg]	Slope	R^2
Increment	15	-0.24	-0.41	-0.08	0.38	0.34	0.994	0.9998
Decrement	9	0.14	0.09	0.19	0.15	0.14	0.996	0.9999
Combined	24	-0.1	-0.23	0.03	0.31	0.27	0.995	1.0000

Table 5.2: Summary statistics for rotation accuracy about the beam axis, corresponding to Fig. 5.14.

Condition	N	Mean [deg]	CI low [deg]	CI high [deg]	RMSE [deg]	MAE [deg]	Slope	R^2
Increment	10	-0.21	-0.24	-0.17	0.21	0.21	0.995	0.9999
Decrement	12	0.33	0.3	0.36	0.33	0.33	0.992	0.9999
Combined	22	0.08	-0.04	0.21	0.28	0.27	0.993	1.0000

Table 5.3: Summary statistics for rotation accuracy about the elevation axis, corresponding to Fig. 5.16.

Condition	N	Mean [deg]	CI low [deg]	CI high [deg]	RMSE [deg]	MAE [deg]	Slope	R^2
Increment	36	-0.12	-0.21	-0.03	0.29	0.22	0.999	1.0000
Decrement	36	0.27	0.19	0.35	0.35	0.3	0.998	1.0000
Combined	72	0.08	0	0.15	0.32	0.26	0.999	1.0000

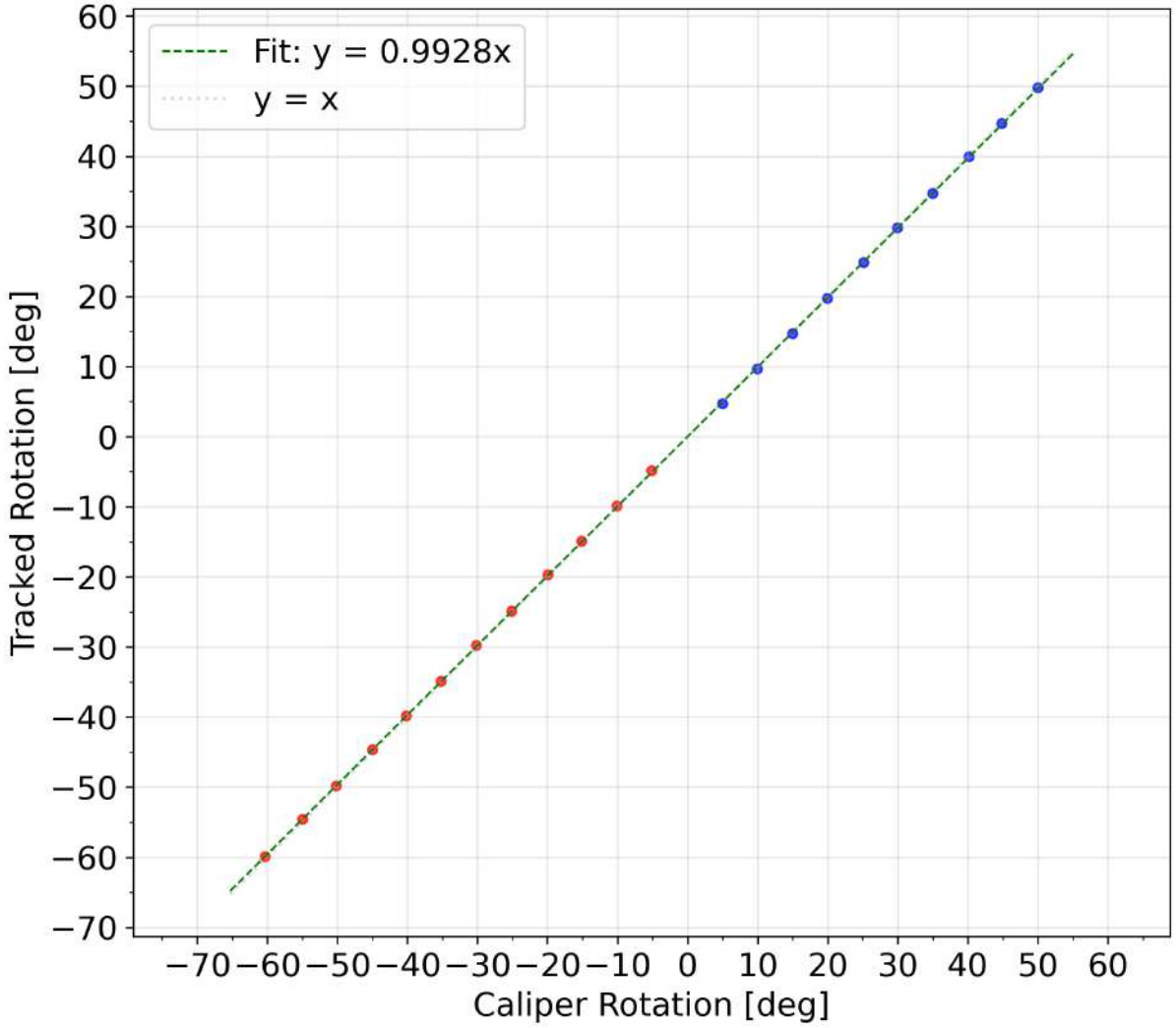


Figure 5.14: Rotation accuracy about the beam axis, with increment (blue) and decrement (red) measurements relative to the caliper reference with the identity relation (gray) and best-fit regression (green).

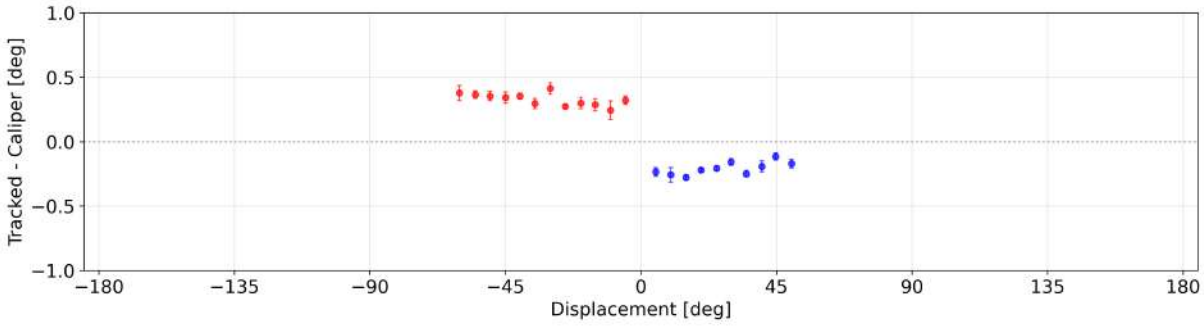


Figure 5.15: Beam rotation error as a function of angular displacement. The points show the mean difference between the algorithm estimate and the caliper reading for each step, and the error bars represent the standard deviation of the fluctuations measured while the system is held still.

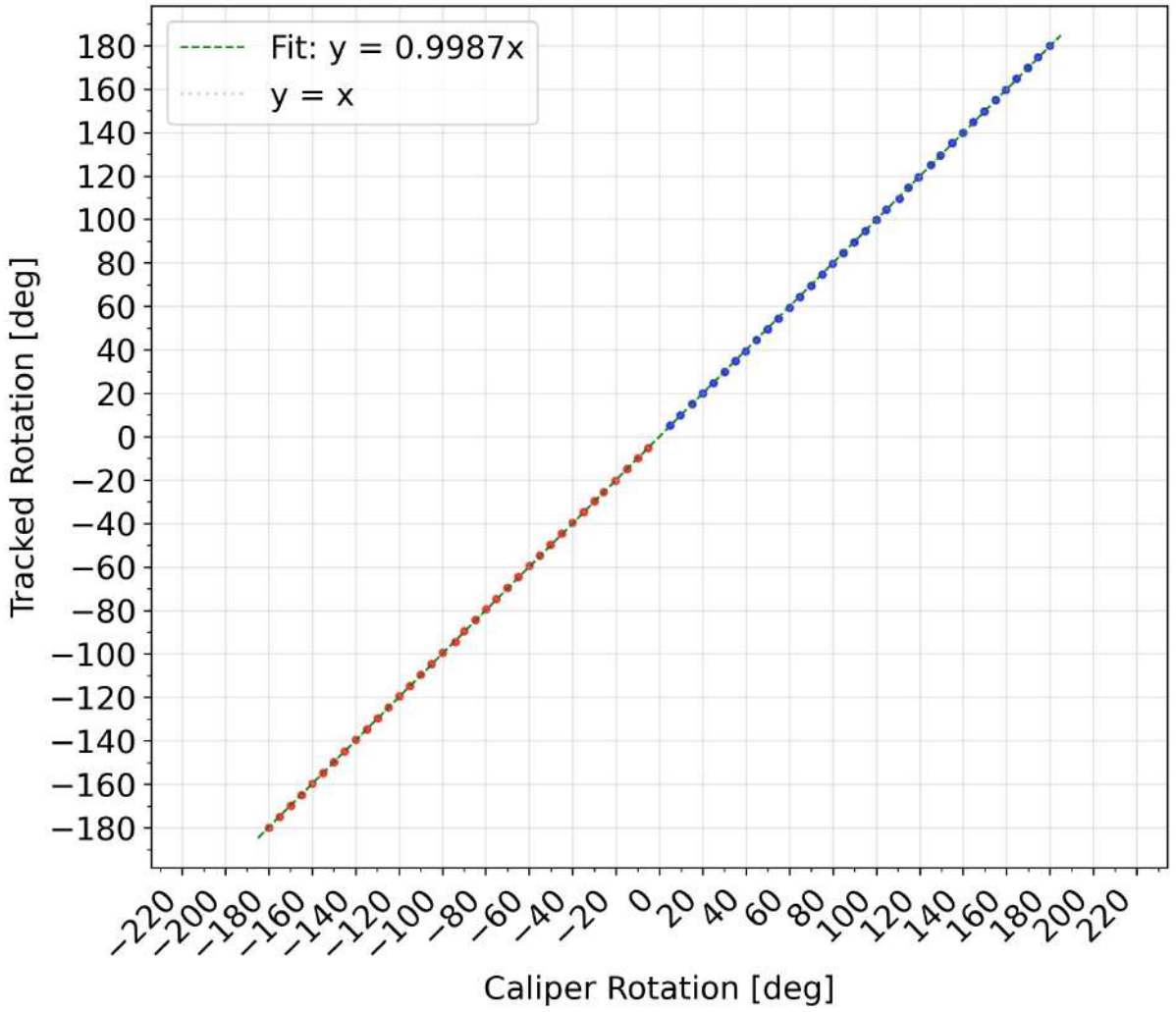


Figure 5.16: Rotation accuracy about the elevation axis, with increment (blue) and decrement (red) measurements relative to the caliper reference with the identity relation (gray) and best-fit regression (green).

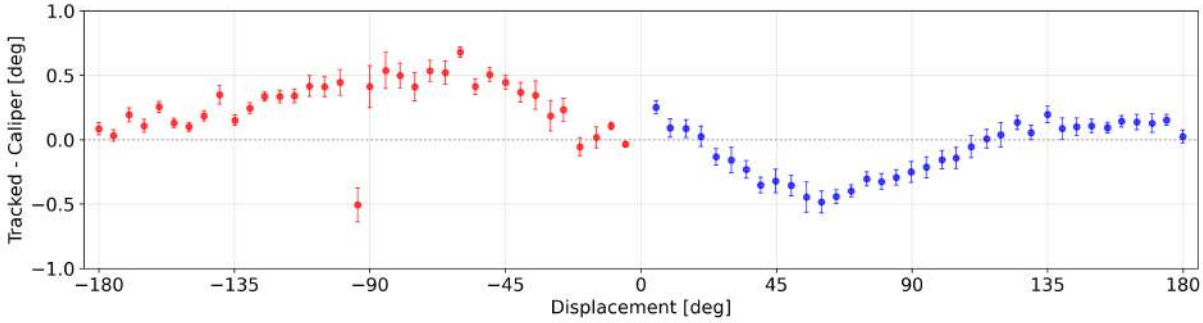


Figure 5.17: Elevation rotation error as a function of angular displacement. The points show the mean difference between the algorithm estimate and the caliper reading for each step, and the error bars represent the standard deviation of the fluctuations measured while the system is held still.

Table 5.4: Summary statistics of rotation tracking error by configuration for Fig. 5.2.

Configuration	N	Mean [deg]	CI low [deg]	CI high [deg]	Std dev [deg]	SEM [deg]	RMSE [deg]	MAE [deg]	Median [deg]	IQR [deg]
Trans 0°-45°	29	−0.71	−0.79	−0.63	0.22	0.04	0.74	0.71	−0.66	0.3
Trans 45°-315°	28	−0.84	−0.86	−0.81	0.07	0.01	0.84	0.84	−0.85	0.12
Beam 0°-45°	29	−1.75	−1.76	−1.73	0.04	0.01	1.75	1.75	−1.75	0.05
Beam 45°-315°	29	−0.55	−0.57	−0.53	0.06	0.01	0.55	0.55	−0.56	0.09
Elev 0°-45°	28	0.62	0.56	0.68	0.16	0.03	0.64	0.62	0.65	0.16
Elev 45°-315°	30	0.54	0.5	0.59	0.13	0.02	0.56	0.54	0.54	0.16

Table 5.5: Summary statistics of rotation tracking error by configuration for Fig. 5.6.

Configuration	N	Mean [deg]	CI low [deg]	CI high [deg]	Std dev [deg]	SEM [deg]	RMSE [deg]	MAE [deg]	Median [deg]	IQR [deg]
Beam 0°-45°	10	−0.61	−0.65	−0.58	0.05	0.02	0.61	0.61	−0.61	0.05

Table 5.6: Summary statistics of rotation tracking error by configuration for Fig. 5.10.

Configuration	N	Mean [deg]	CI low [deg]	CI high [deg]	Std dev [deg]	SEM [deg]	RMSE [deg]	MAE [deg]	Median [deg]	IQR [deg]
Trans 0°-45°	9	−0.42	−0.47	−0.37	0.06	0.02	0.43	0.42	−0.41	0.03
Trans 45°-315°	9	−0.76	−0.86	−0.65	0.14	0.05	0.77	0.76	−0.74	0.2
Beam 0°-45°	10	−0.44	−0.49	−0.39	0.07	0.02	0.44	0.44	−0.44	0.08
Beam 45°-315°	10	−0.52	−0.57	−0.47	0.07	0.02	0.52	0.52	−0.5	0.13
Elev 0°-45°	10	−0.37	−0.47	−0.27	0.15	0.05	0.39	0.37	−0.41	0.11
Elev 45°-315°	10	−0.87	−0.92	−0.83	0.06	0.02	0.88	0.87	−0.87	0.07

5.2 Translational Error

5.2.1 Translation Performance with the Final 6×10 Design

Translational accuracy was assessed using the same 6×10 HydraMarker configuration as in the final rotational tests (Fig. 5.8 (b)). The probe prototype was mounted on the linear stage and 30 translations were performed along each axis as shown in Fig. 4.5. The virtual marker map from the final rotational tests (Fig. 5.9) was reused for pose estimation. The resulting translational error distributions and mean-subtracted repeatability results are reported in Figs. 5.18 and 5.19, with Table 5.13 summarizing the corresponding configuration-level statistics. Across all axes, the measured translations exhibited a systematic bias of approximately $1.5\,\mathrm{mm}$ in magnitude (positive for transverse and beam, negative for elevation), clearly indicating a dominant calibration or reconstruction error component rather than pure random noise.

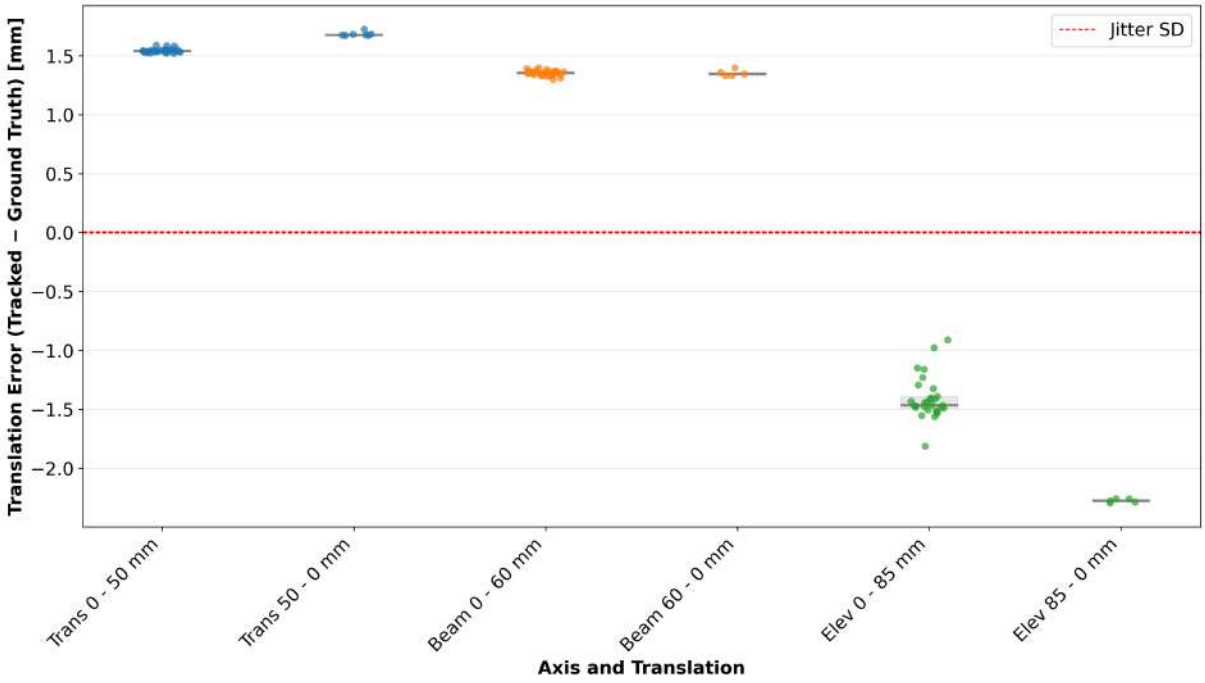


Figure 5.18: Translational tracking error for the three axes (transverse in blue, beam in orange, elevation in green) evaluated over different ranges. Each point is an individual trial and gray median/IQR overlays summarize each configuration. Red dashed lines indicate the translational jitter standard deviation reference level ($0.005\,\mathrm{mm}$).

Accuracy tests were performed using incremental translation sequences along each axis, similar to the rotational accuracy protocol. Shown in Figs. 5.20 – 5.22, the tracked translation error relative to the linear stage position is plotted for increment and decrement

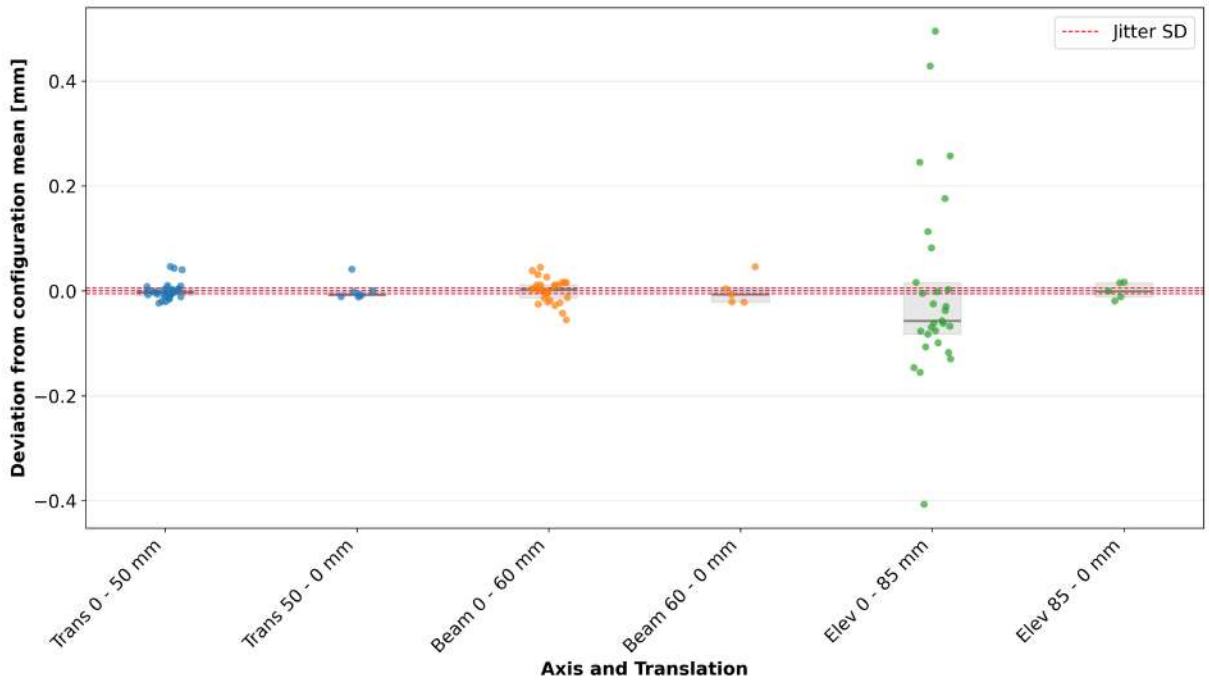


Figure 5.19: Mean-subtracted translational tracking error for Fig. 5.18, obtained by subtracting the mean error within each axis/range subset. Points show trial-wise deviations from the configuration mean and gray median/IQR overlays summarize spread. Red dashed lines indicate the jitter SD reference level from the static test ($0.005\,\mathrm{mm}$).

trials, with slope-only least-squares fits included to capture systematic drift. Here, the increment trial moves from the initial position to the end of the travel range, while the decrement trial is the return to the initial position, with $10\,\mathrm{mm}$ steps. The accompanying tables 5.7– 5.9 report relevant statistics.

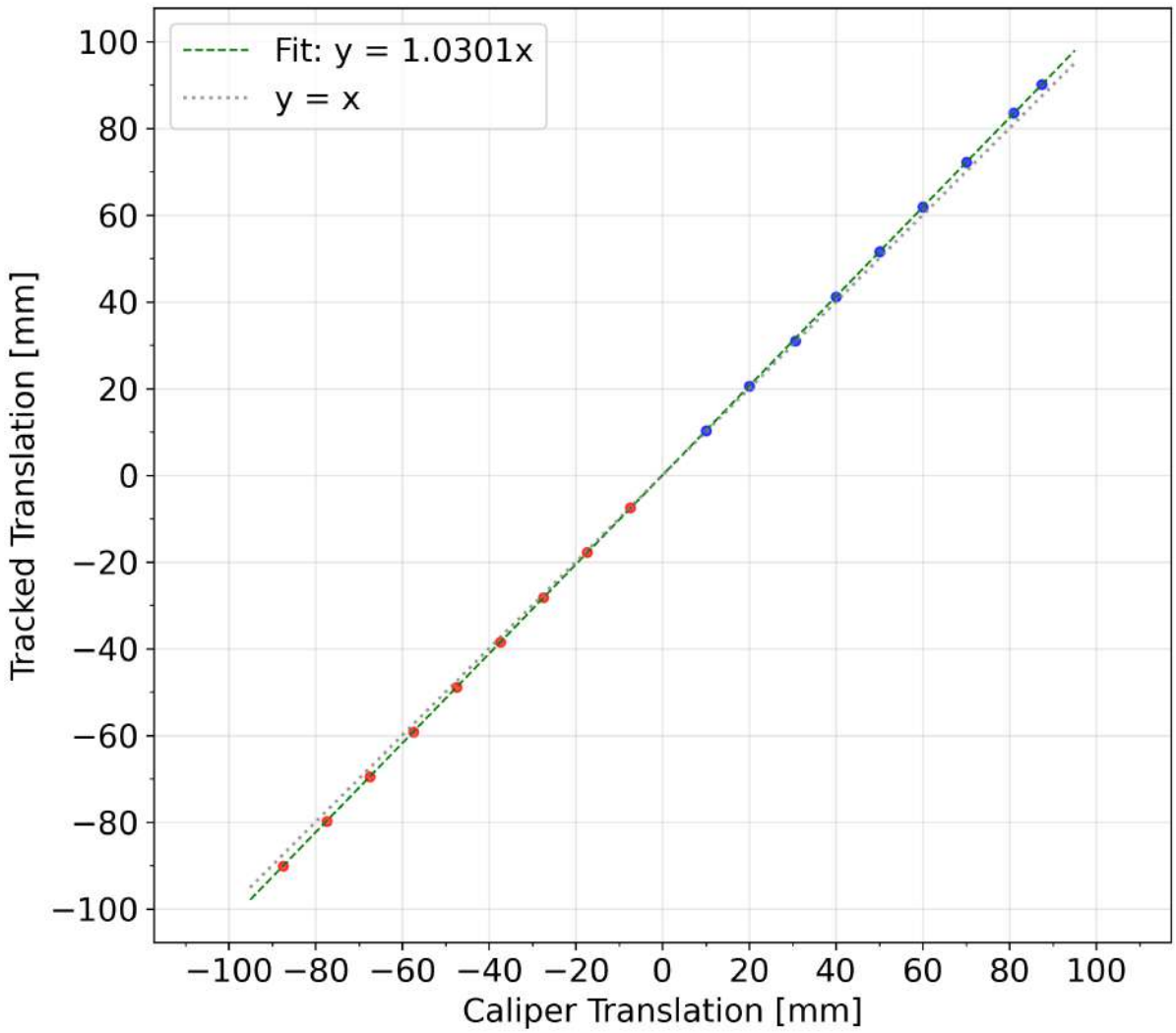


Figure 5.20: Translation accuracy along the transverse axis, with increment (blue) and decrement (red) measurements relative to the caliper reference with the identity relation (gray) and best-fit regression (green).

Table 5.7: Summary statistics for translation accuracy along the transverse axis, corresponding to Fig. 5.20.

Condition	N	Mean [mm]	CI low [mm]	CI high [mm]	RMSE [mm]	MAE [mm]	Slope	R^2
Increment	9	1.43	0.69	2.17	1.7	1.43	1.030	0.9999
Decrement	9	−1.4	−2.09	−0.71	1.64	1.4	1.030	1.0000
Combined	18	0.02	−0.84	0.87	1.67	1.42	1.030	1.0000

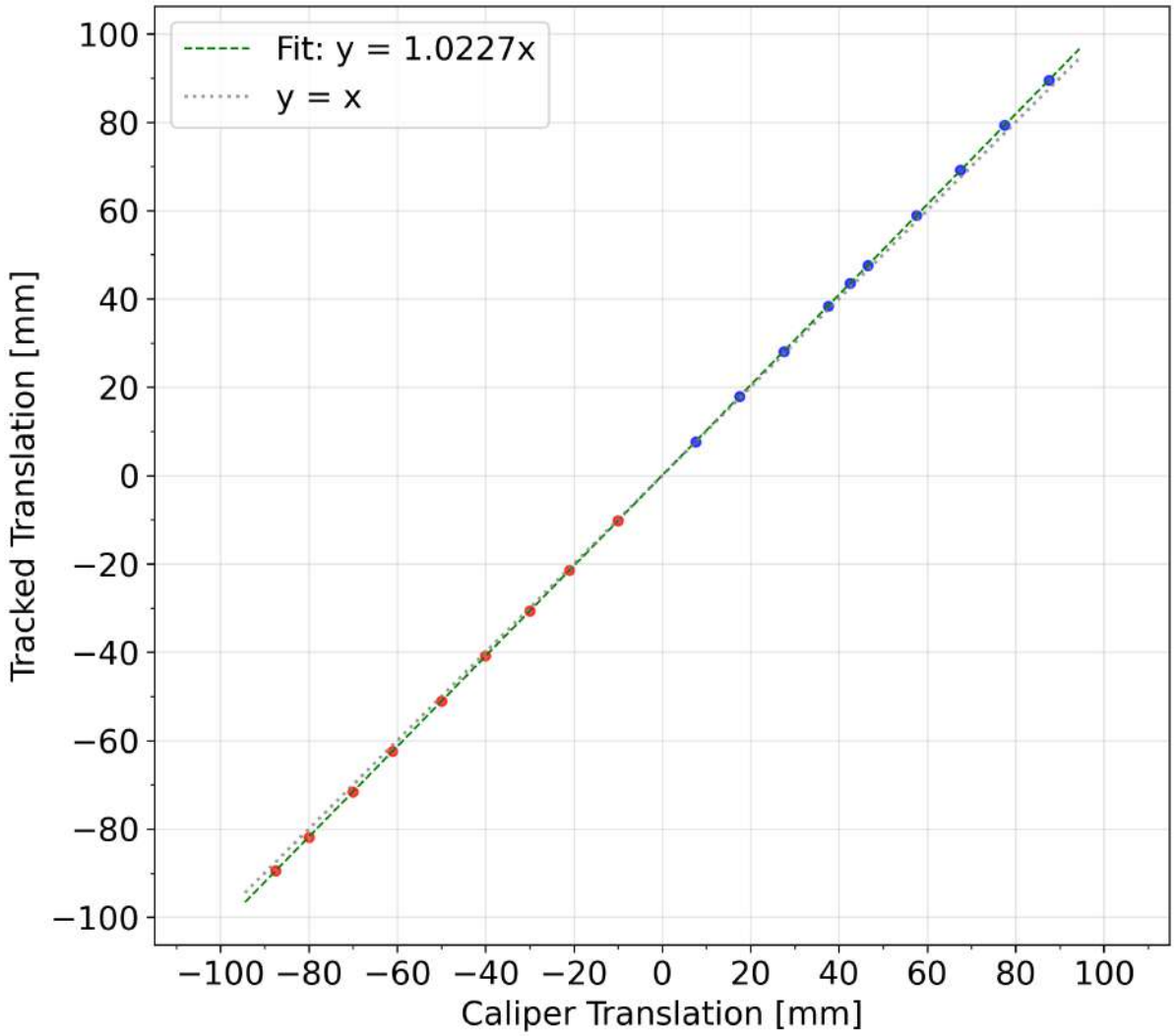


Figure 5.21: Translation accuracy about the beam axis, with increment (blue) and decrement (red) measurements relative to the caliper reference with the identity relation (gray) and best-fit regression (green).

Table 5.8: Summary statistics for translation accuracy along the beam axis, corresponding to Fig. 5.21.

Condition	N	Mean [mm]	CI low [mm]	CI high [mm]	RMSE [mm]	MAE [mm]	Slope	R^2
Increment	10	1.07	0.62	1.52	1.22	1.07	1.023	1.0000
Decrement	9	−1.08	−1.59	−0.58	1.25	1.08	1.022	1.0000
Combined	19	0.05	−0.56	0.66	1.23	1.08	1.023	1.0000

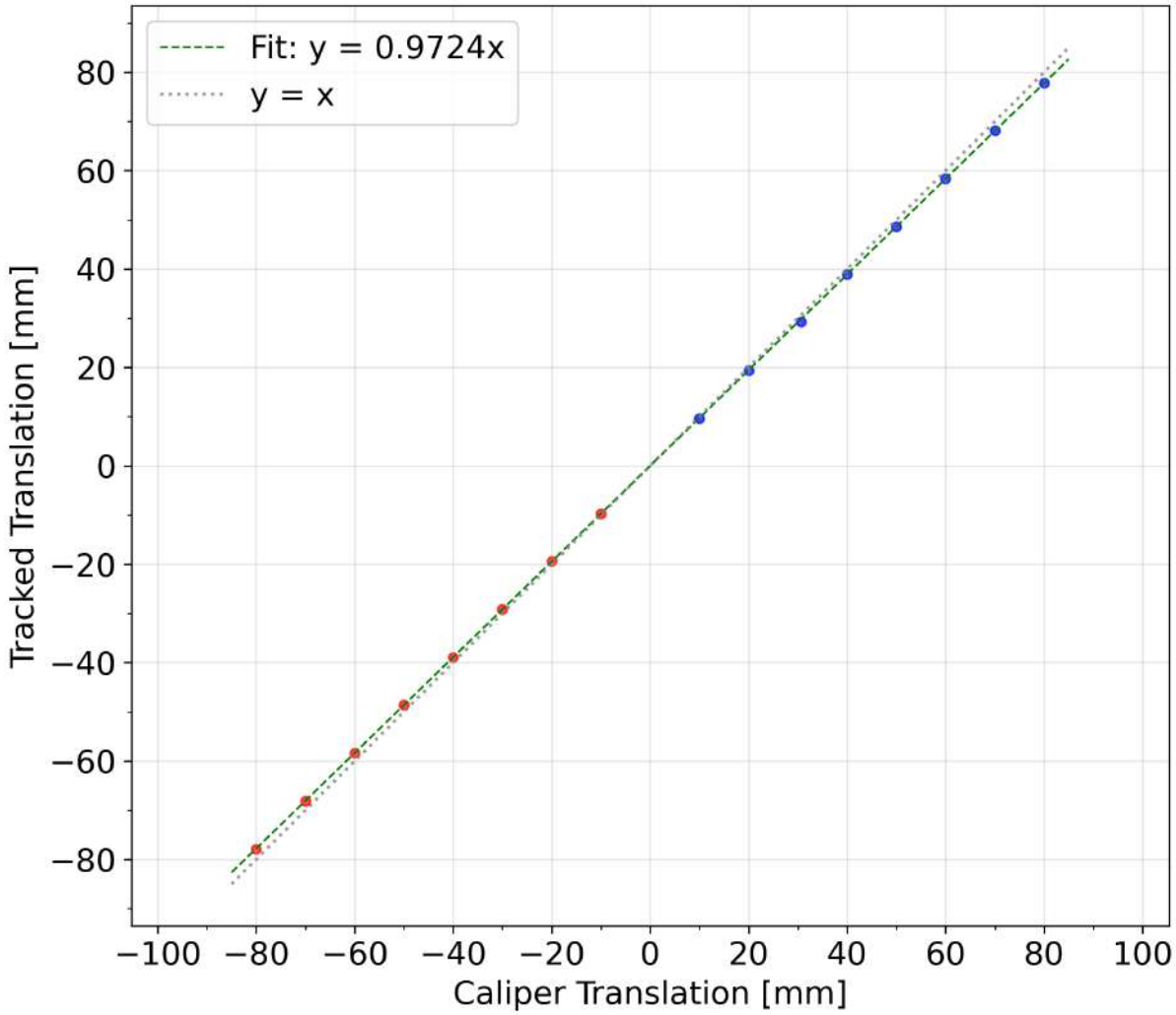


Figure 5.22: Translation accuracy along the elevation axis, with increment (blue) and decrement (red) measurements relative to the caliper reference with the identity relation (gray) and best-fit regression (green).

Table 5.9: Summary statistics for translation accuracy along the elevation axis, corresponding to Fig. 5.22.

Condition	N	Mean [mm]	CI low [mm]	CI high [mm]	RMSE [mm]	MAE [mm]	Slope	R^2
Increment	8	−1.3	−1.82	−0.78	1.43	1.3	0.972	0.9999
Decrement	8	1.24	0.7	1.77	1.37	1.24	0.973	1.0000
Combined	16	−0.03	−0.8	0.74	1.4	1.27	0.972	1.0000

5.2.2 Translation Performance after Camera Re-calibration

Given the pronounced bias observed in the results and the error growth with distance, the experiments were repeated using a re-calibrated camera and an updated virtual model (post-BA reprojection error 0.072 px). The results are shown in Figs. 5.23 – 5.24, with

statistics reported in Table 5.14.

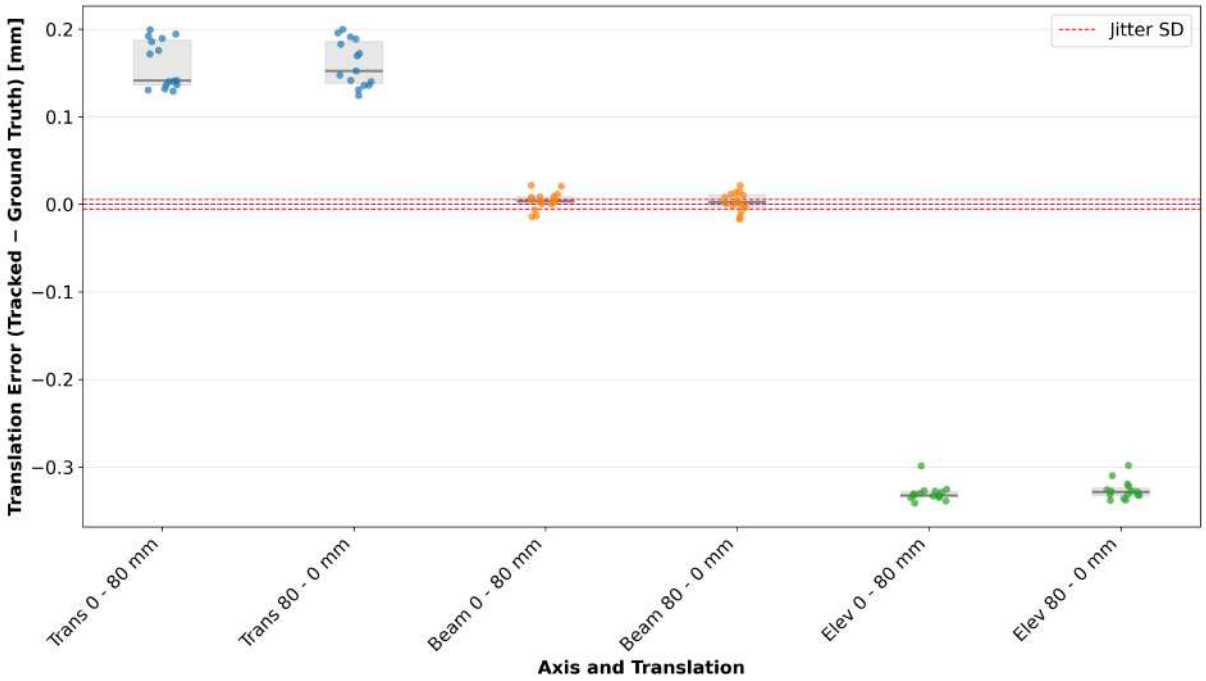


Figure 5.23: Translational tracking error for the three axes (transverse in blue, beam in orange, elevation in green) over $80\,\mathrm{mm}$ ranges after re-calibration. Each point is an individual trial and gray median/IQR overlays summarize each configuration. Red dashed lines indicate the translational jitter standard deviation reference level ($0.005\,\mathrm{mm}$).

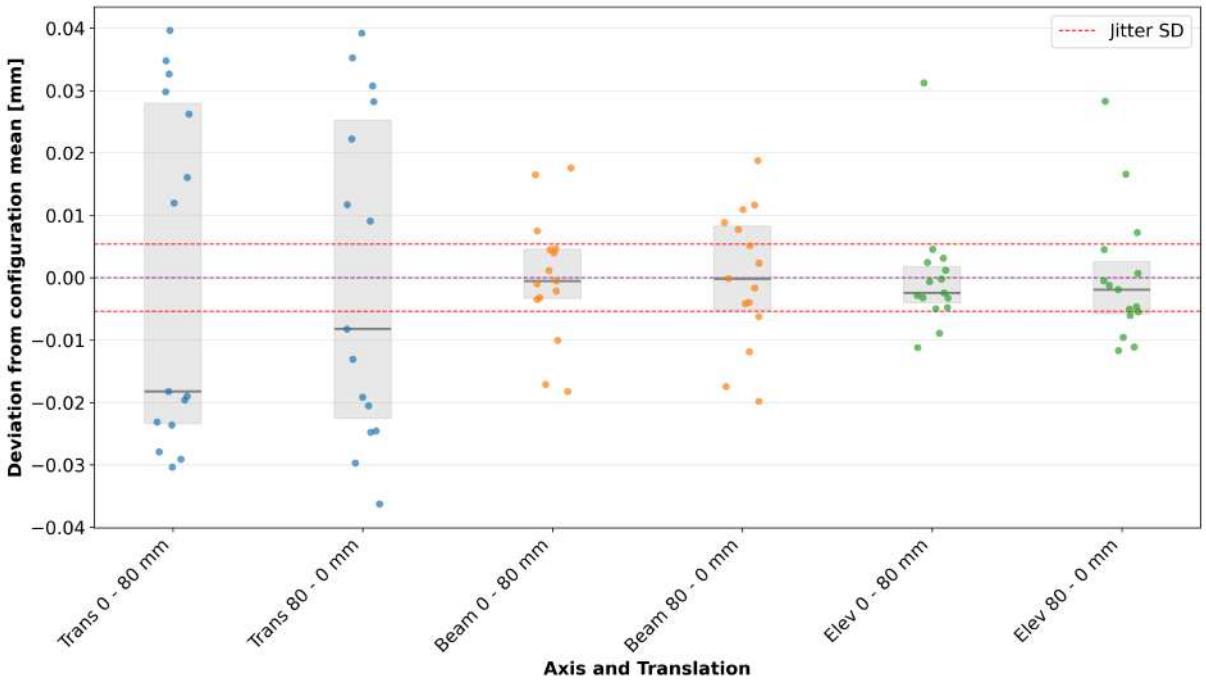


Figure 5.24: Mean-subtracted translational tracking error for Fig. 5.23, obtained by subtracting the mean error within each axis/range subset. Points show trial-wise deviations from the configuration mean and gray median/IQR overlays summarize spread. Red dashed lines indicate the jitter SD reference level from the static test ($0.005\,\mathrm{mm}$).

The accuracy experiments were also repeated after re-calibration, with results shown in Figs. 5.25 – 5.29. These trials follow a different protocol than the previous ones, instead of starting from one end of the translation range, the probe was initially positioned at the center of the stage travel and then moved to one end, where $10\,\mathrm{mm}$ increments started towards the other end. The difference was then calculated relative to the starting central position. The corresponding statistics are reported in Tables 5.10– 5.12. Representative error profiles for transverse, axial, and elevational translations are shown in Figs. 5.26–5.30, where each point denotes the mean difference between tracked and caliper displacement at a given step and error bars indicate jitter variability.

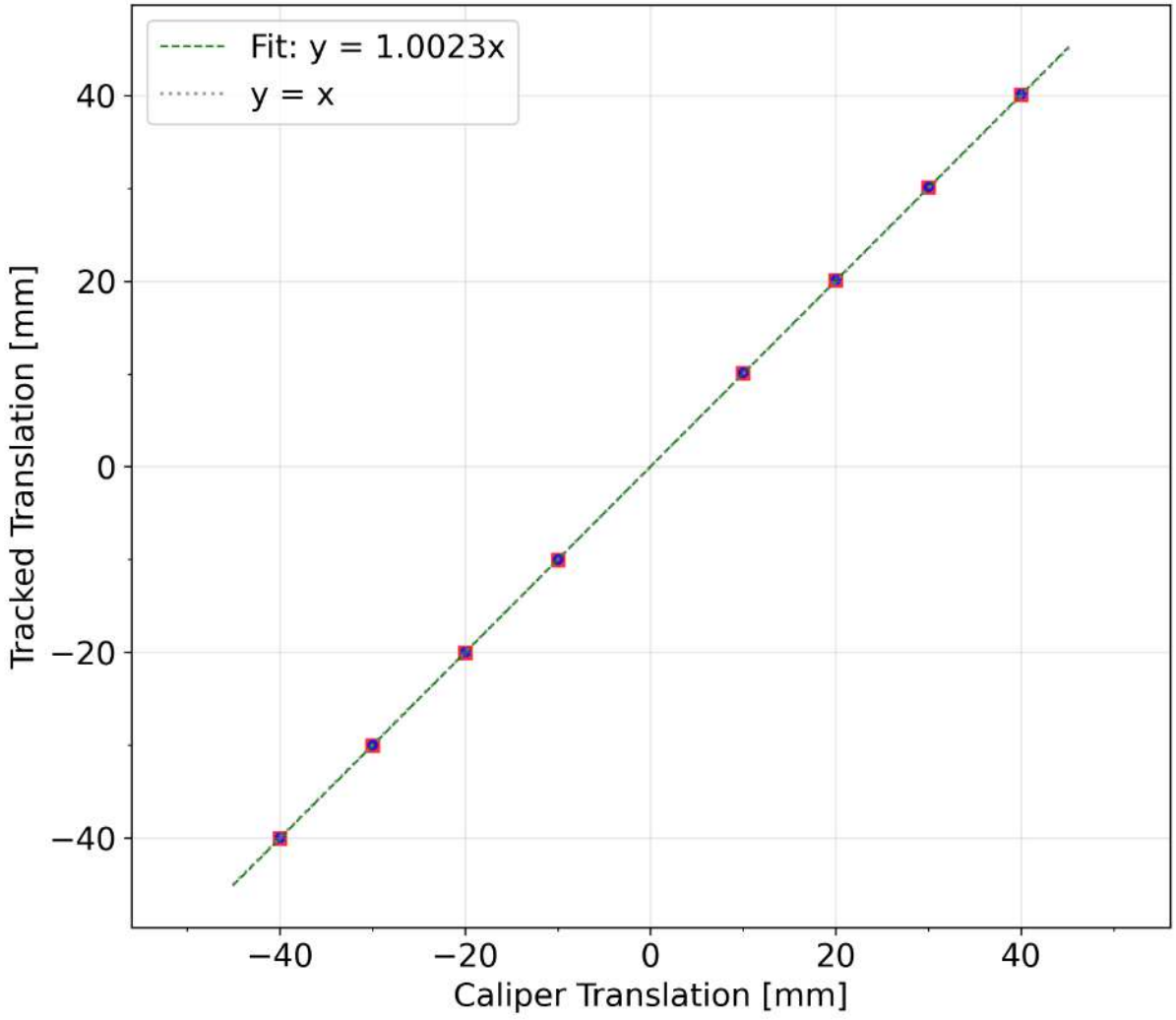


Figure 5.25: Translation accuracy along the transverse axis of two repeated trials (red squares and blue circles) relative to the caliper reference with the identity relation (gray) and best-fit regression (green) after re-calibration.

Fig. 5.27 includes a single unintended sample at $0\,\mathrm{mm}$. Despite this, the return-to-start repeatability is high, with an initial-to-return difference of $0.002\,\mathrm{mm}$.

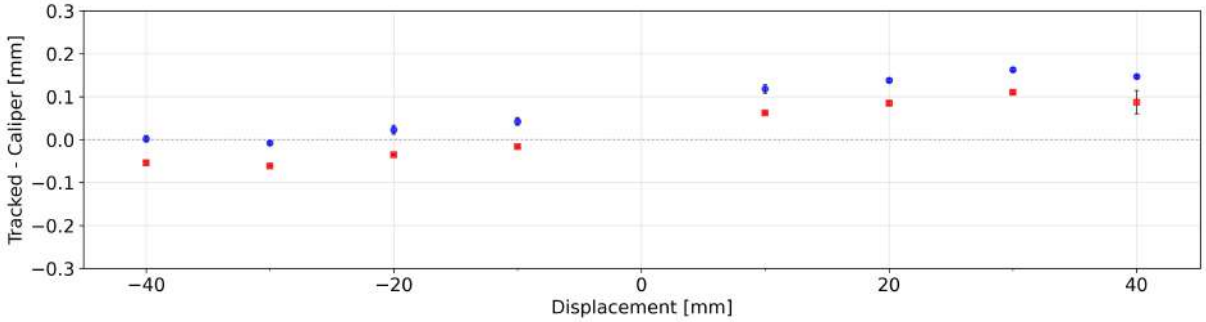


Figure 5.26: Transverse translation error as a function of angular displacement. The points show the mean difference between the algorithm estimate and the caliper reading for each step, and the error bars represent the standard deviation of the fluctuations measured while the system is held still.

Table 5.10: Summary statistics for translation accuracy along the transverse axis, after re-calibration, corresponding to Fig. 5.25.

Condition	N	Mean [mm]	CI low [mm]	CI high [mm]	RMSE [mm]	MAE [mm]	Slope	R^2
Trial 1	8	0.02	−0.04	0.08	0.07	0.06	1.002	1.0000
Trial 2	8	0.08	0.02	0.14	0.1	0.08	1.002	1.0000
Combined	16	0.05	0.01	0.09	0.09	0.07	1.002	1.0000

Table 5.11: Summary statistics for translation accuracy along the beam axis, after re-calibration, corresponding to Fig. 5.27.

Condition	N	Mean [mm]	CI low [mm]	CI high [mm]	RMSE [mm]	MAE [mm]	Slope	R^2
Trial 1	8	−0.01	−0.04	0.02	0.03	0.02	0.999	1.0000
Trial 2	9	−0.01	−0.03	0.01	0.03	0.02	0.999	1.0000
Combined	17	−0.01	−0.03	0.01	0.03	0.02	0.999	1.0000

Table 5.12: Summary statistics for translation accuracy along the elevation axis, after re-calibration, corresponding to Fig. 5.29.

Condition	N	Mean [mm]	CI low [mm]	CI high [mm]	RMSE [mm]	MAE [mm]	Slope	R^2
Trial 1	8	0.06	−0.04	0.16	0.12	0.11	0.996	1.0000
Trial 2	8	−0.02	−0.12	0.07	0.11	0.09	0.996	1.0000
Combined	16	0.02	−0.05	0.08	0.12	0.1	0.996	1.0000

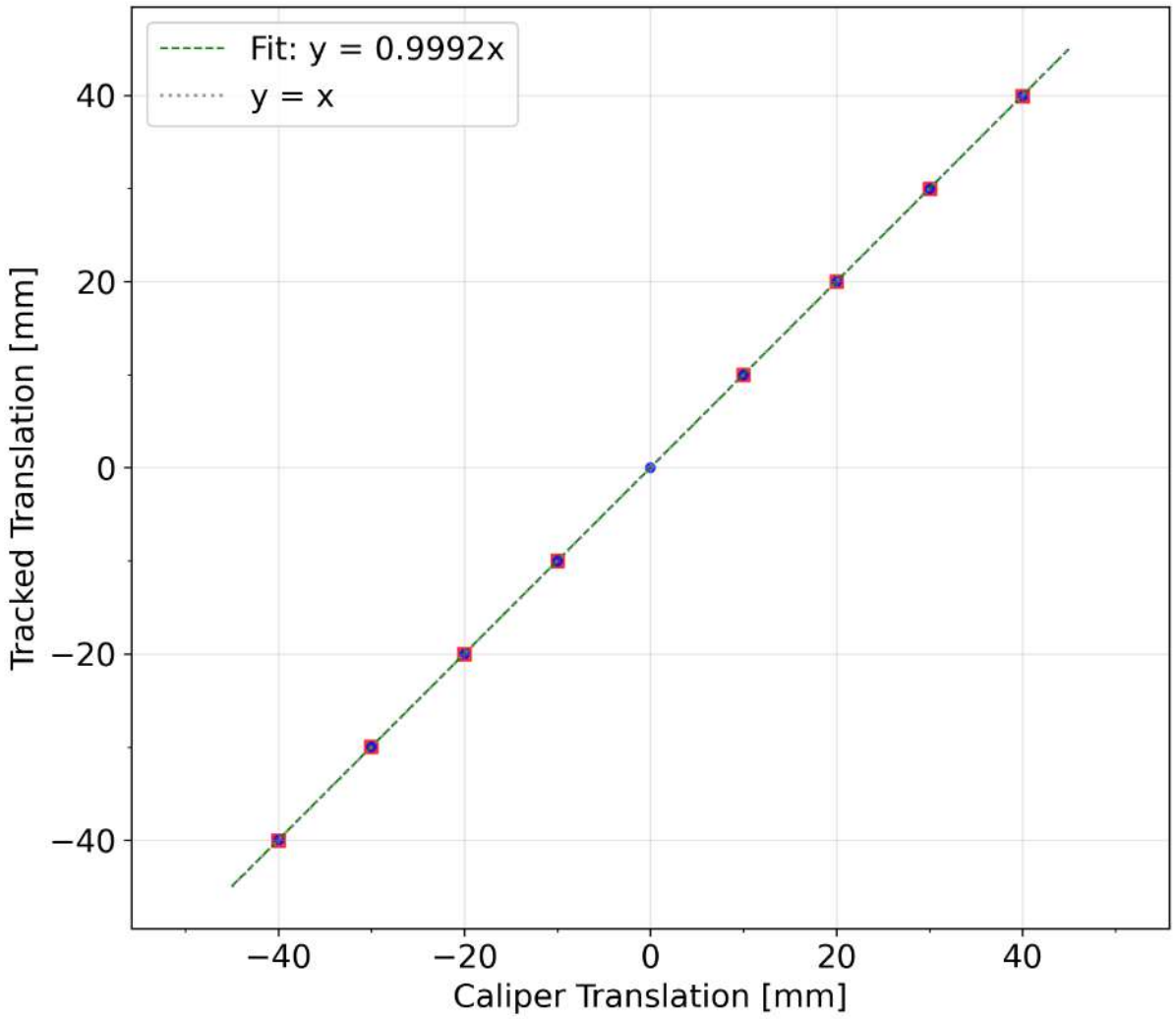


Figure 5.27: Translation accuracy along the beam axis of two repeated trials (red squares and blue circles) relative to the caliper reference with the identity relation (gray) and best-fit regression (green) after re-calibration.

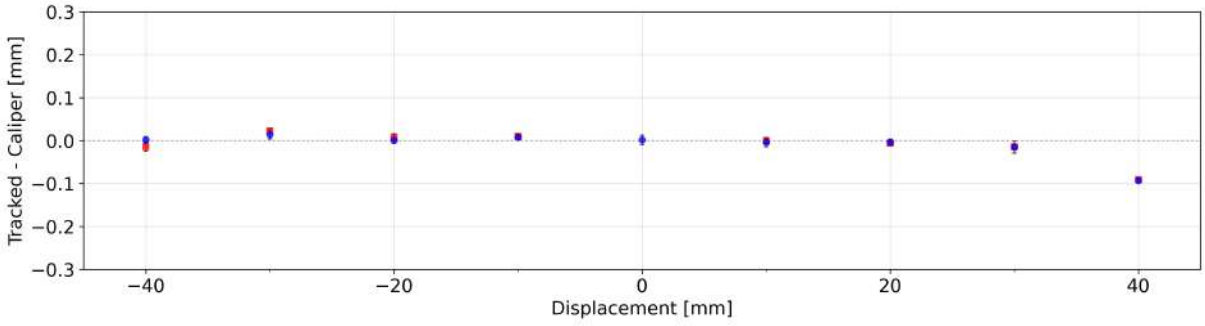


Figure 5.28: Beam translation error as a function of angular displacement. The points show the mean difference between the algorithm estimate and the caliper reading for each step, and the error bars represent the standard deviation of the fluctuations measured while the system is held still.

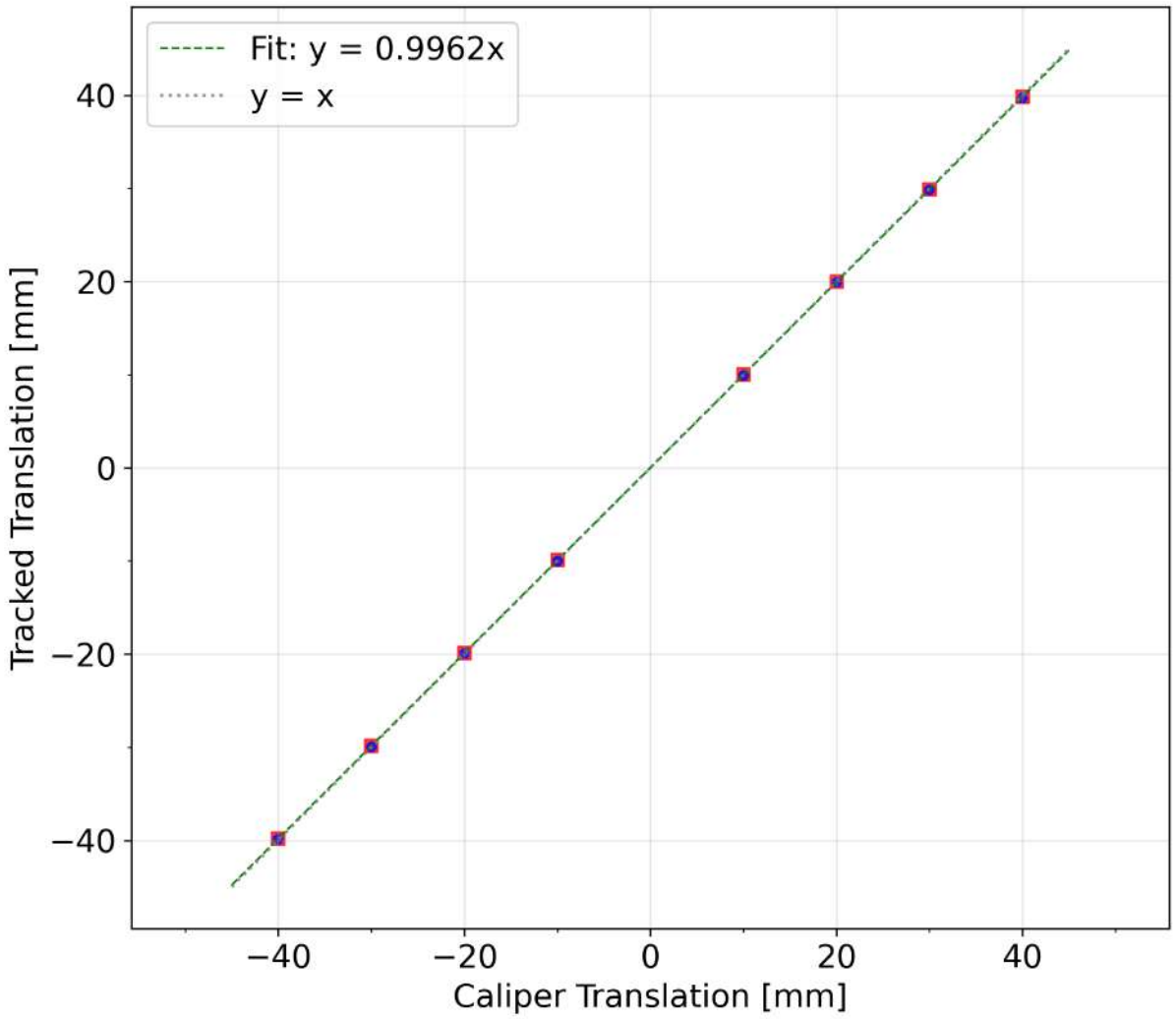


Figure 5.29: Translation accuracy along the elevation axis of two repeated trials (red squares and blue circles) relative to the caliper reference with the identity relation (gray) and best-fit regression (green) after re-calibration.

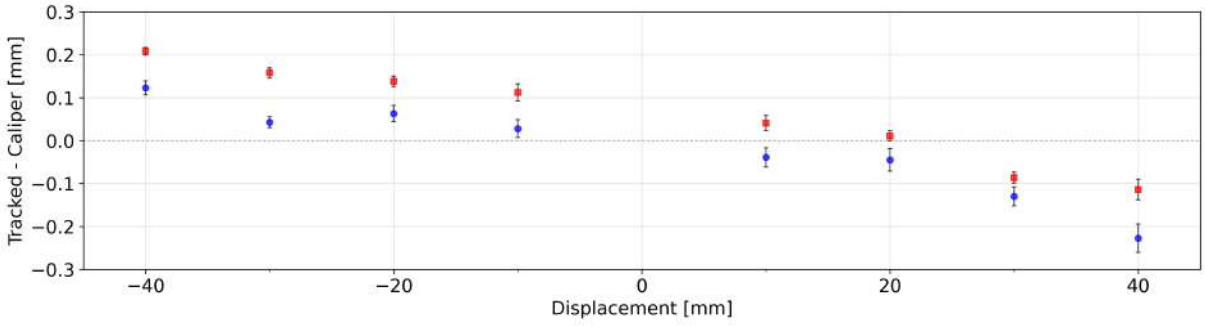


Figure 5.30: Elevation translation error as a function of angular displacement. The points show the mean difference between the algorithm estimate and the caliper reading for each step, and the error bars represent the standard deviation of the fluctuations measured while the system is held still.

5.2.3 Verification of Transverse-Axis Error Distribution

Since the translational error along the transverse axis is non-normal and bimodal (see Fig. 5.23), these trials were repeated. The results are reported in Figs. 5.31 – 5.32, with statistics in Table 5.15.

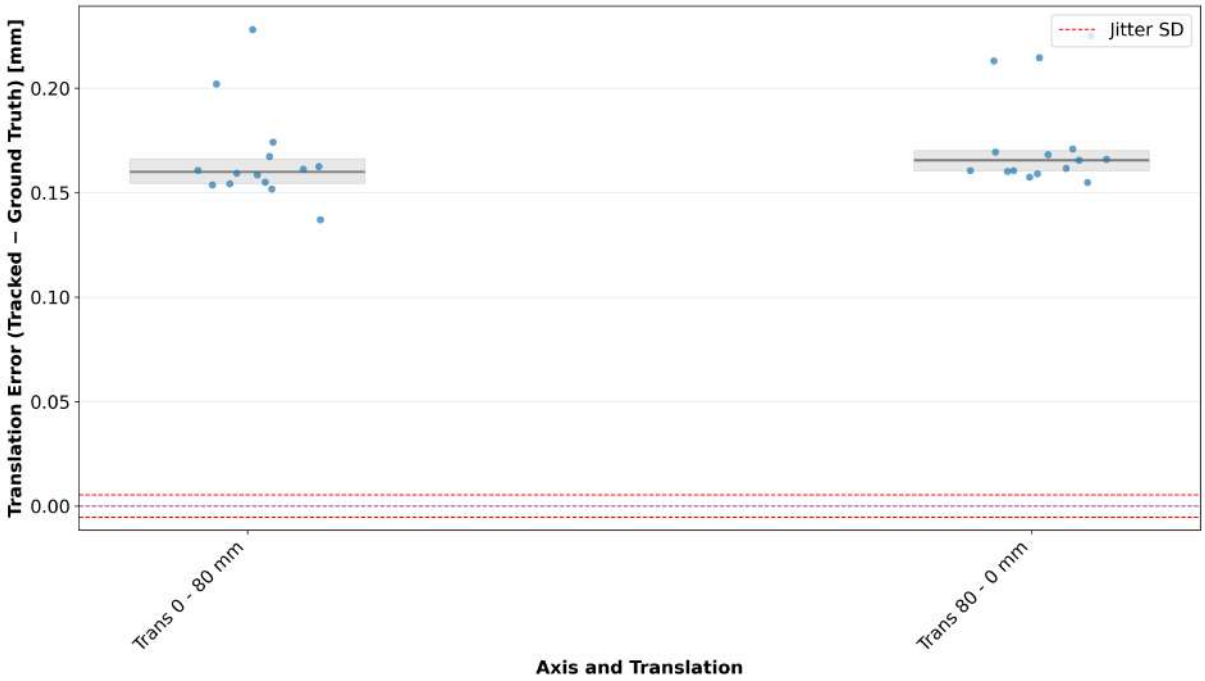


Figure 5.31: Translational tracking error for the transverse axis over $80\,\mathrm{mm}$ range. Each point is an individual trial and gray median/IQR overlays summarize each configuration. Red dashed lines indicate the translational jitter standard deviation reference level ($0.005\,\mathrm{mm}$).

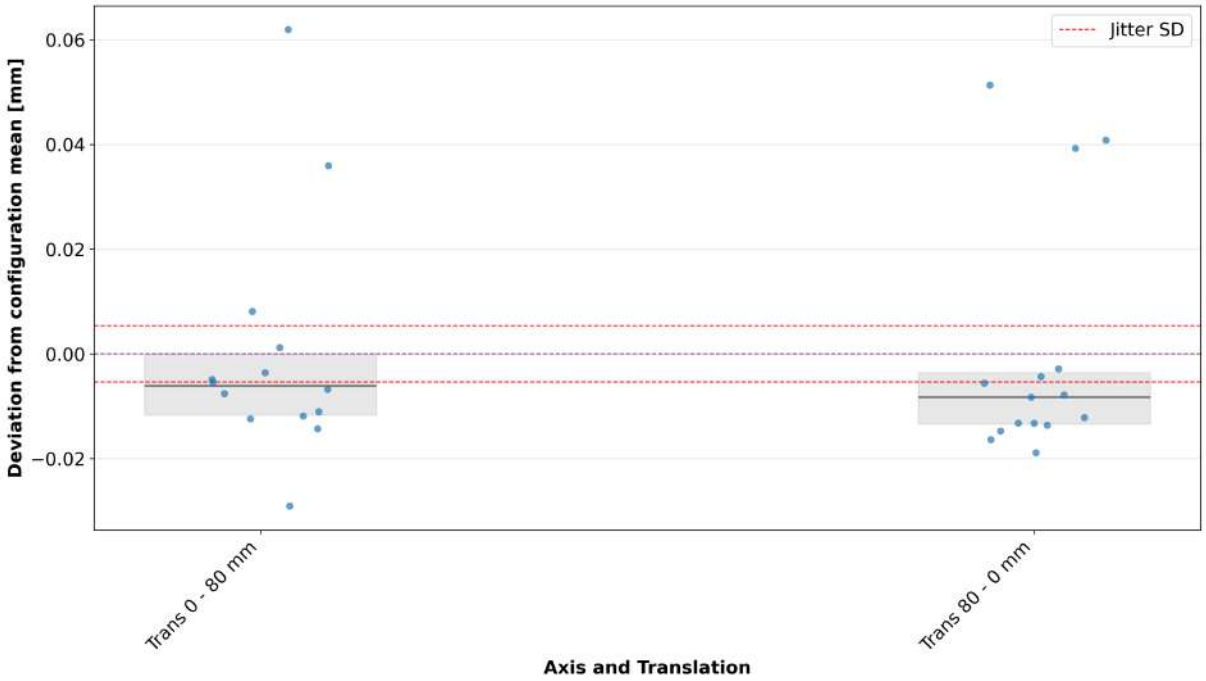


Figure 5.32: Mean-subtracted translational tracking error for Fig. 5.31, obtained by subtracting the mean error within each axis/range subset. Points show trial-wise deviations from the configuration mean and gray median/IQR overlays summarize spread. Red dashed lines indicate the jitter SD reference level from the static test ($0.005\,\mathrm{mm}$).

Table 5.13: Summary statistics of translational tracking error by configuration for Fig. 5.18.

Configuration	N	Mean [mm]	CI low [mm]	CI high [mm]	Std dev [mm]	SEM [mm]	RMSE [mm]	MAE [mm]	Median [mm]	IQR [mm]
Trans 0 - 50 mm	31	1.5	1.5	1.5	0.02	0	1.5	1.5	1.5	0.01
Trans 50 - 0 mm	7	1.7	1.7	1.7	0.02	0.01	1.7	1.7	1.7	0.01
Beam 0 - 60 mm	29	1.4	1.3	1.4	0.02	0	1.4	1.4	1.4	0.03
Beam 60 - 0 mm	5	1.4	1.3	1.4	0.03	0.01	1.4	1.4	1.3	0.03
Elev 0 - 85 mm	29	−1.4	−1.5	−1.3	0.18	0.03	1.4	1.4	−1.5	0.1
Elev 85 - 0 mm	5	−2.3	−2.3	−2.3	0.02	0.01	2.3	2.3	−2.3	0.03

Table 5.14: Summary statistics of translational tracking error, after re-calibration, by configuration for Fig. 5.23.

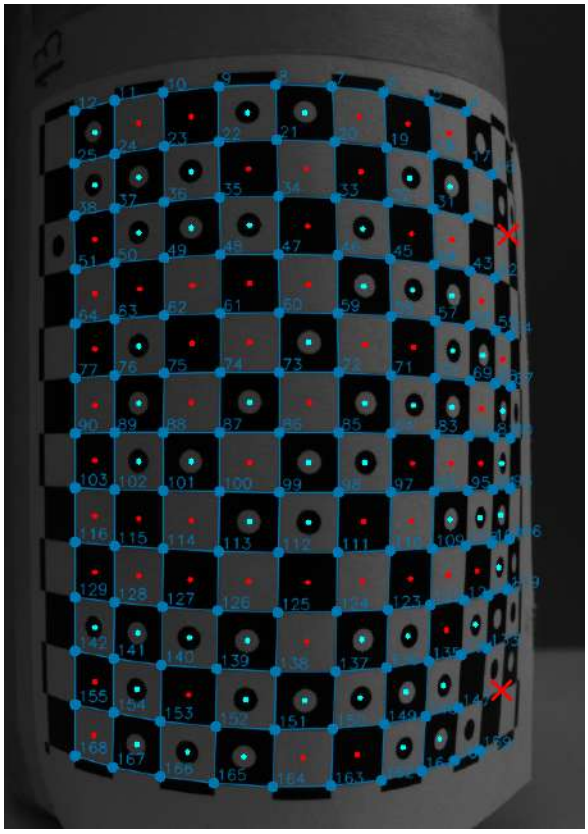
Configuration	N	Mean [mm]	CI low [mm]	CI high [mm]	Std dev [mm]	SEM [mm]	RMSE [mm]	MAE [mm]	Median [mm]	IQR [mm]
Trans 0 - 80 mm	15	0.2	0.1	0.2	0.03	0.01	0.2	0.2	0.1	0.05
Trans 80 - 0 mm	15	0.2	0.1	0.2	0.03	0.01	0.2	0.2	0.2	0.05
Beam 0 - 80 mm	15	0	0	0	0.01	0	0	0	0	0.01
Beam 80 - 0 mm	15	0	0	0	0.01	0	0	0	0	0.01
Elev 0 - 80 mm	15	−0.3	−0.3	−0.3	0.01	0	0.3	0.3	−0.3	0.01
Elev 80 - 0 mm	15	−0.3	−0.3	−0.3	0.01	0	0.3	0.3	−0.3	0.01

Table 5.15: Summary statistics of translational tracking error by configuration for Fig. 5.31.

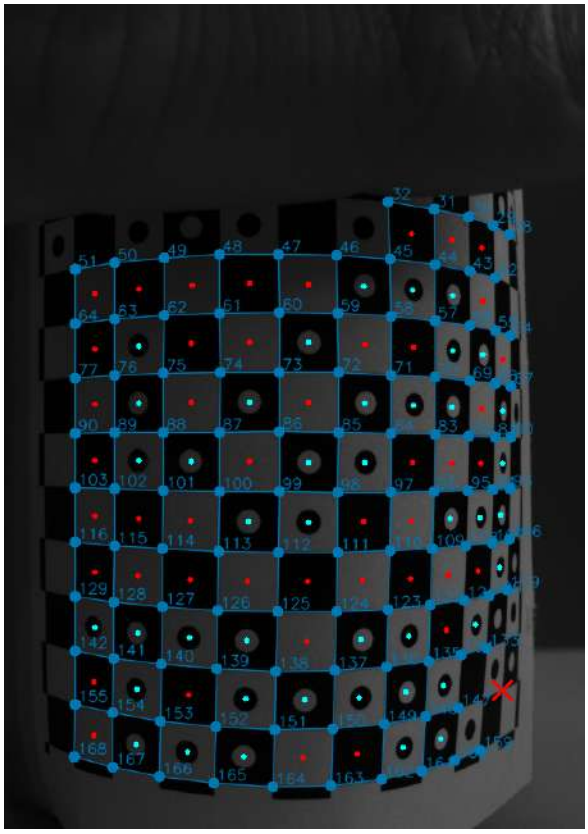
Configuration	N	Mean [mm]	CI low [mm]	CI high [mm]	Std dev [mm]	SEM [mm]	RMSE [mm]	MAE [mm]	Median [mm]	IQR [mm]
Trans 0 - 80 mm	14	0.2	0.2	0.2	0.02	0.01	0.2	0.2	0.2	0.01
Trans 80 - 0 mm	15	0.2	0.2	0.2	0.02	0.01	0.2	0.2	0.2	0.01

5.3 Single-Marker Occlusion Tolerance

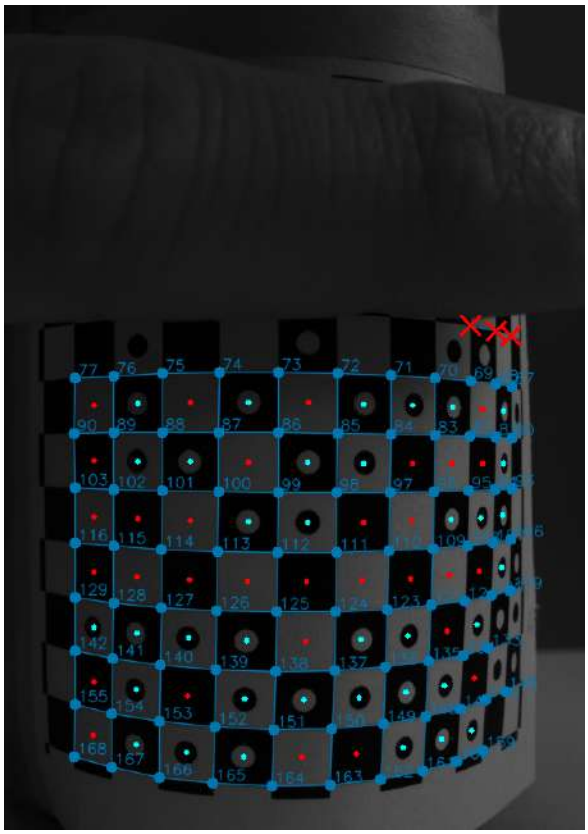
Progressive occlusion testing of the 12×12 HydraMarker with 4×4 ID tags revealed that the marker remained identifiable even with approximately 91.67% of its area occluded (Figure 5.33 (a–g)). Specifically, successful identification was maintained until only a single horizontal line (approximately 8.33% of the marker) remained visible (Figure 5.33 (g)). Beyond this threshold, the identification algorithm failed to correctly decode the marker ID. In Figure 5.33, detected corners are displayed as dark blue dots if successfully identified, or red crosses if not. Within each cell, cyan and red dots indicate the sampling results: cyan denotes successful sampling with contrast above the threshold (Section 3.4.5), while red indicates failed sampling below it.



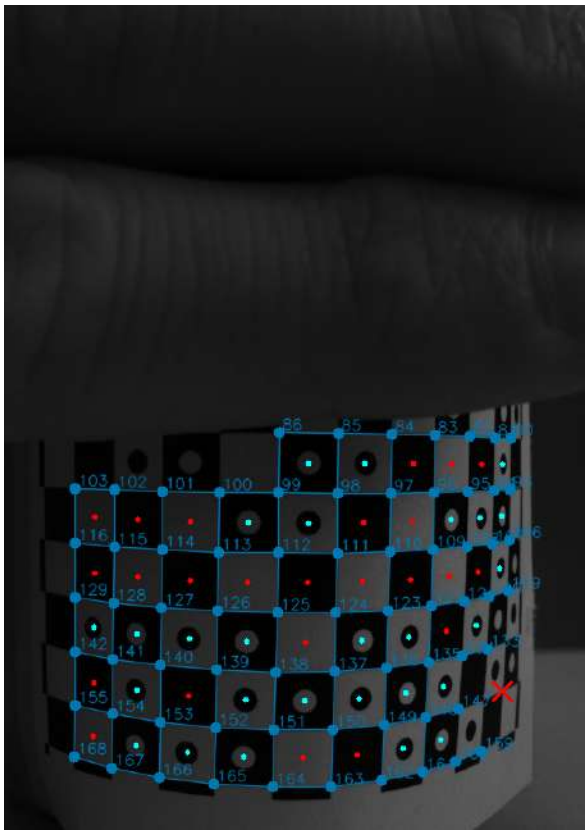
(a)



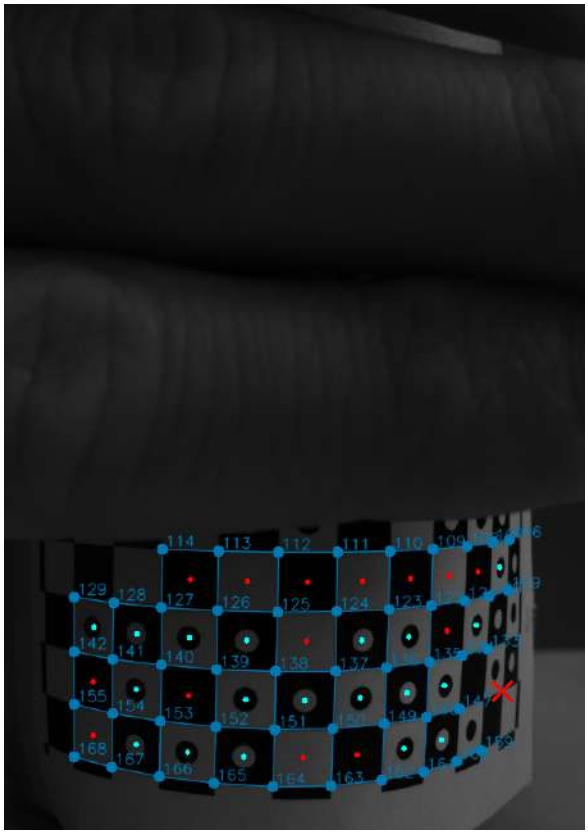
(b)



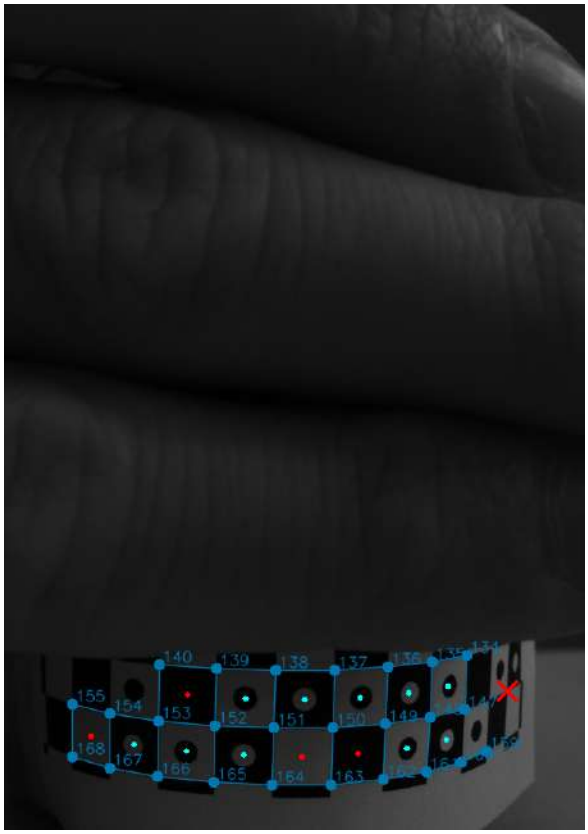
(c)



(d)



(e)



(f)

(g)

Figure 5.33: Occlusion robustness of the HydraMarker: example frames (a–g) showing progressive hand occlusion with successful corner detection and identification across varying visible marker extents.

To prevent false identifications under extreme occlusion, a confidence threshold was implemented: if fewer than 15% of the total available IDs within the marker field matrix were successfully detected, the identification was rejected and marked as invalid. Failed detections are displayed as orange cells and crosses in the overlay, as shown in Figure 5.33 (g). This conservative approach prioritizes identification reliability over detection rate, ensuring that only high-confidence pose estimates are propagated to subsequent processing stages.

5.4 Dual-Marker Detection and Identification

Simultaneous detection and identification of two distinct 10×10 HydraMarker fields was successfully demonstrated across multiple test frames. The two markers are shown in Figure 5.34. The optimized identification algorithm correctly assigned detected grids to their corresponding marker fields without false cross-assignments.

Figures 5.35 and 5.36 illustrate this capability: both images capture the same camera frame, but with overlays isolating the identification results for Marker A and Marker B respectively. In each frame, one marker is displayed with successfully detected corners (blue dots), while the other is marked with failed detections (red crosses), demonstrating the algorithm's robust ability to disambiguate multiple distinct markers and prevent erroneous cross-identifications.

Figure 5.34: Marker A (left) and Marker B (right): two 10×10 HydraMarker fields with 3×3 and 2×5 ID tags, mounted on cylindrical surfaces.

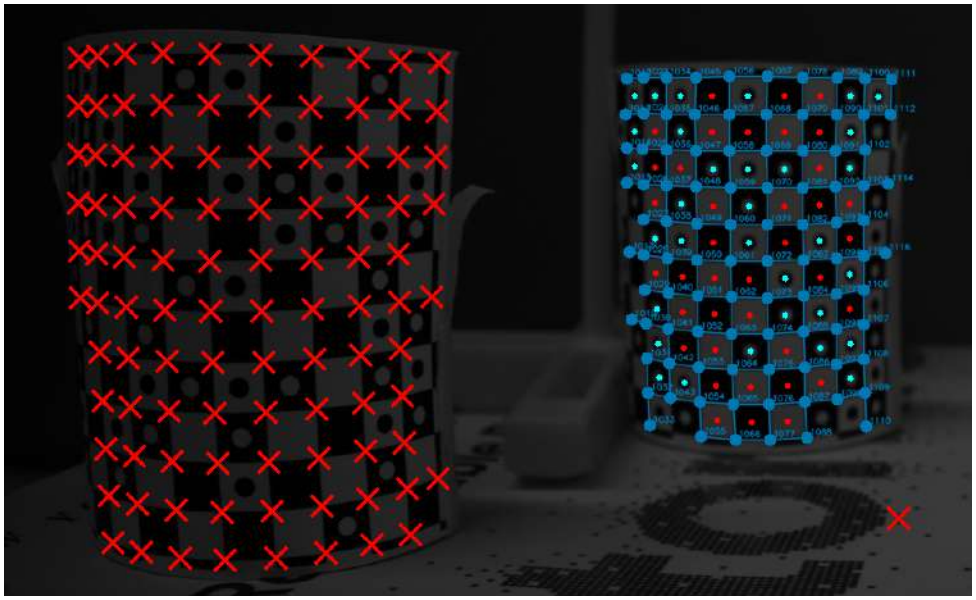


Figure 5.35: Identification of Marker A from dual-marker frame: detected corners highlighted in blue, while unidentified Marker B shown as red crosses.

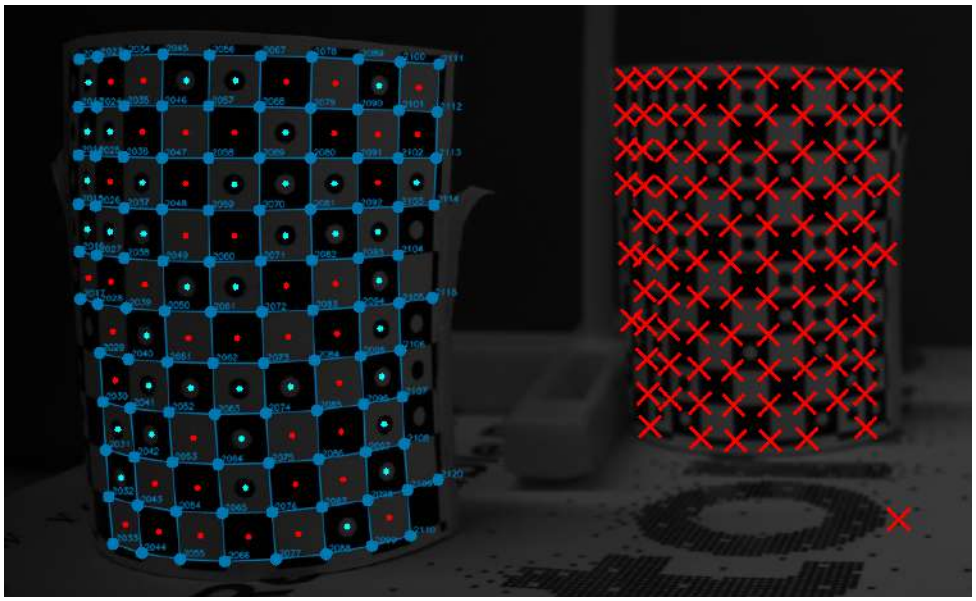


Figure 5.36: Identification of Marker B from dual-marker frame: detected corners highlighted in blue, while unidentified Marker A shown as red crosses.

Furthermore, to validate rejection capability, the algorithm was tested by attempting to match Marker A's field definition against a frame containing only Marker B. As shown in Figure 5.37, the algorithm correctly failed to assign any ID to Marker A, with all attempted matches marked as failed detections (red crosses). This confirms that the identification algorithm does not generate false positives and reliably rejects markers that do not correspond to the queried field definition.

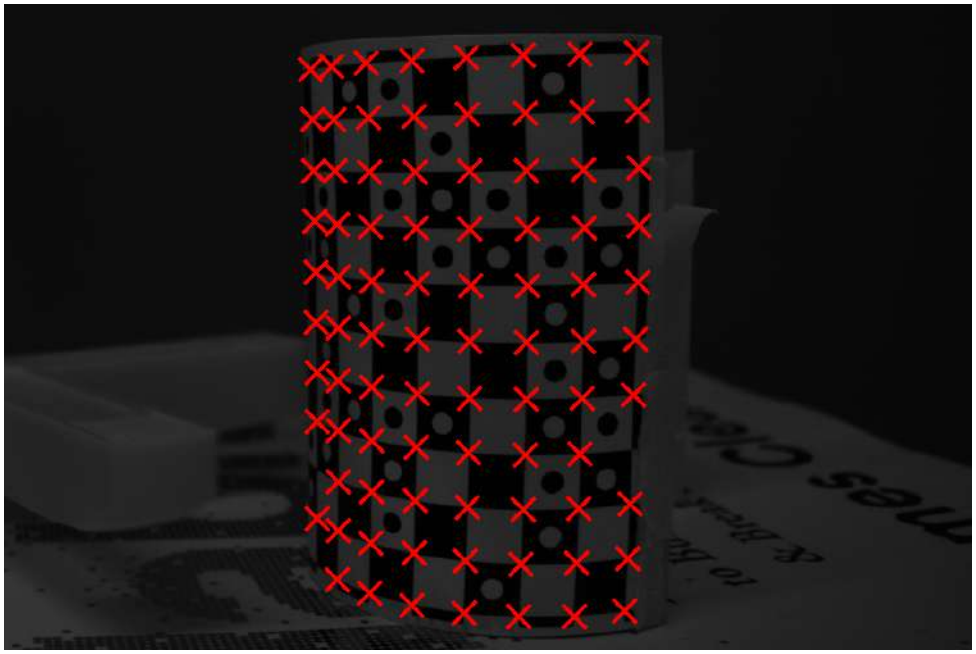


Figure 5.37: Failed attempt to match Marker A field definition against Marker B.

5.5 Marker Integration on Clarius L7 Probe

The validated marker was printed on a laminated vinyl sticker to improve durability and resistance to disinfectants. The sticker was applied to the Clarius L7 probe and the assembly was placed on a turntable to perform the mapping step, as shown in Fig. 5.38. The resulting virtual map after bundle adjustment is shown in Fig. 5.39 and achieved a reprojection error of 0.0787 px.

Figure 5.38: 6×10 HydraMarker attached to Clarius L7 ultrasound probe.

Following the procedure described in Section 3.5.6, the marker to probe transform was obtained using the NDI Polaris Vega, as shown in Fig. 5.40. After digitization of the marker corner fiducials and composition of the transformation matrices, the 3D virtual map was successfully overlaid on the Clarius probe CAD model (Fig. 5.41). The resulting transform matrix is:

$$ {}^{M}\mathbf{T}_{P} = \begin{bmatrix} 0.965 & 0.028 & -0.006 & 46.8 \\ 0.029 & 0.964 & -0.022 & 120.6 \\ 0.005 & 0.022 & 0.965 & 6.0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.1} $$

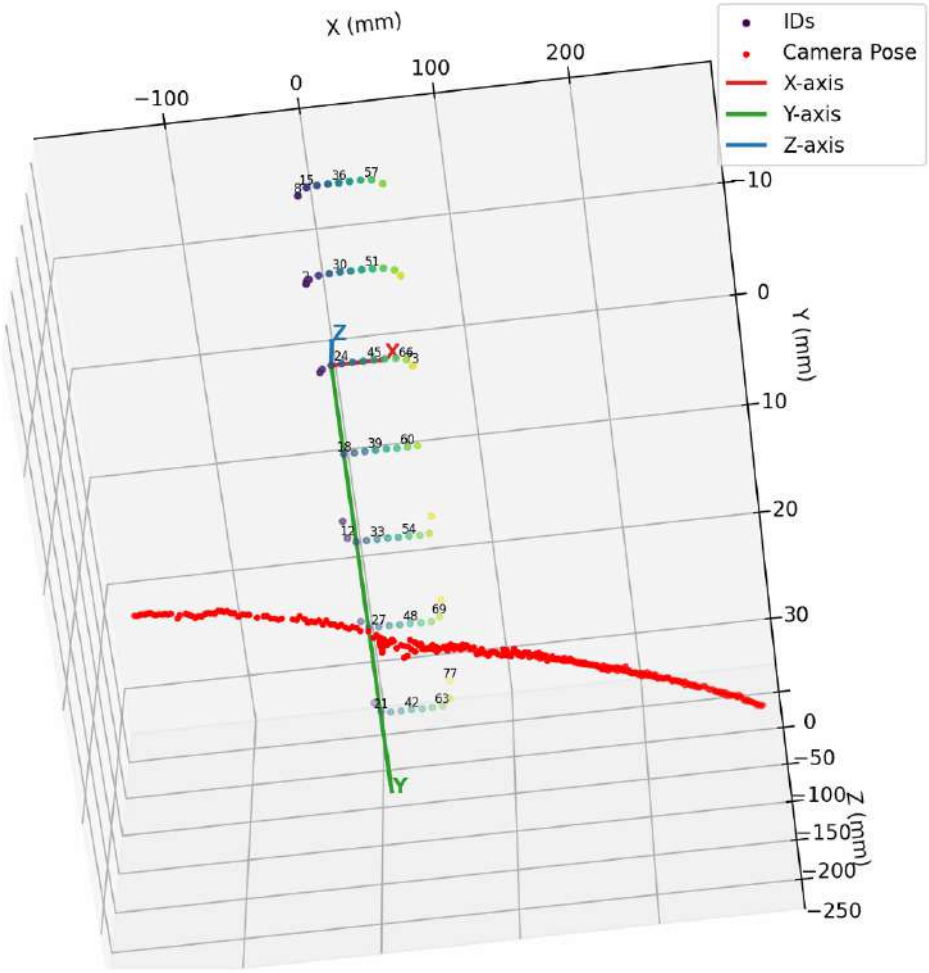


Figure 5.39: Structure from Motion reconstruction of the 6×10 HydraMarker used for pose estimation. Marker corner points are shown with a viridis colormap and estimated camera poses are shown in red. The marker reference frame is indicated by the axes (X red, Y green, Z blue).

Figure 5.40: Digitization setup with the NDI Polaris Vega (left) and the Clarius probe clamped on the right.

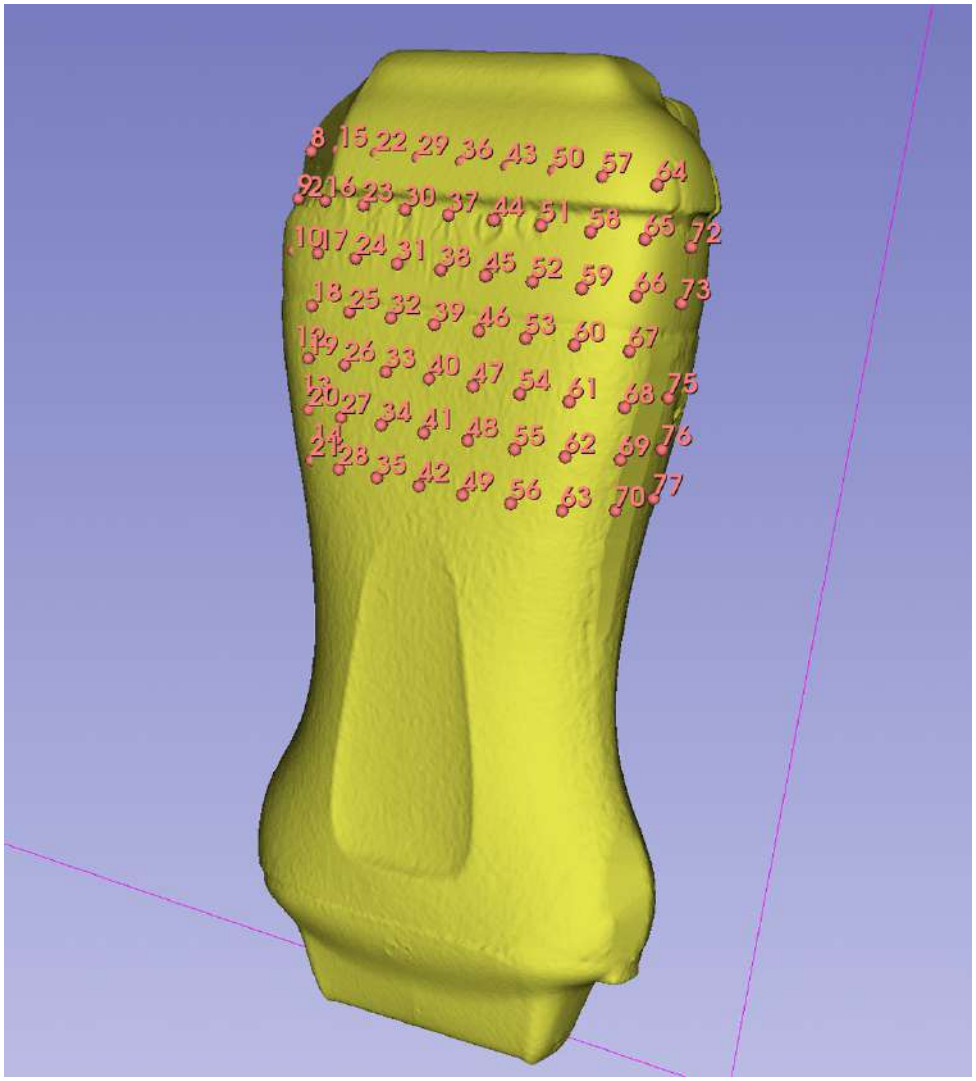


Figure 5.41: Structure from Motion map overlaid on the Clarius probe CAD model in 3D Slicer after fiducials digitization.

Chapter 6

Future Work and Clinical Translation Pathway

6.1 Patient Marker Refinement and Motion Compensation

As introduced in Section 3.2.2, the Outside-In tracking architecture can be extended with a second HydraMarker attached to the patient to monitor motion during acquisition. The patient marker is intended to provide an independent estimate of infant motion relative to the fixed camera reference frame. In a first phase, this information can be used to flag scans affected by motion artifacts. In a second phase, provided sufficient tracking accuracy is achieved, the same information can be used to compensate for patient motion during ultrasound volume reconstruction.

6.1.1 Design Considering Anatomical and Environmental Constraints

The patient marker design was driven primarily by anatomical and handling constraints specific to neonatal hip scanning. A key requirement was reproducible placement with respect to palpable bony landmarks, so that the marker position remained stable relative to the infant pelvis rather than the overlying soft tissues. In particular, the device was conceived to reference the iliac crest and lateral pelvic contour, providing a consistent location for the patterned region across examinations and operators. By anchoring the marker to these landmarks, the design aimed to reduce sensitivity to soft tissue motion

artifacts caused by breathing, muscle activity, or manual handling, which would otherwise introduce relative motion between the skin and the underlying bone.

To preserve marker stability and ensure reliable detection and identification, the surface bearing the marker pattern was supported by a rigid carrier coupled to a compliant bottom layer that conformed to the infant's body surface. The HydraMarker itself was intended to be printed on a vinyl sticker and applied to the rigid carrier. This multilayer construction allowed the patterned area to follow the global motion of the pelvis while limiting local deformation of the marker, thereby mitigating soft tissue-induced errors in the estimated patient motion.

From a tracking perspective, the main requirement is correctness and repeatability of marker detection and identification under representative clinical conditions. The marker layout is therefore conceived as a limited patterned strip (approximately $20\,\mathrm{mm}$ wide and $70\,\mathrm{mm}$ long) distributed along one long side of the annulus and partially along the short sides. This geometry aims to keep the patterned area visible while avoiding interference with the scanning window and reducing the likelihood that the sonographer's hand occludes the marker during the sweep. The resulting constraints introduce a trade-off between maximizing the usable patterned area for robust identification and minimizing the device size. These should be resolved by iterating the carrier geometry and the marker placement until reliable tracking is achieved without compromising clinical usability.

6.1.2 Validation

Once a candidate design is defined, validation shall follow a protocol analogous to the probe-marker evaluation. In the initial phase, the objective is motion flagging rather than high-precision tracking. Therefore, sub-millimetric and sub-degree accuracy are not strictly required at first, provided that motion beyond a relevant threshold can be detected consistently and without false identifications. The motion threshold itself cannot be fixed a priori, as it depends on the sensitivity of the ultrasound volume reconstruction pipeline to patient motion. Consequently, threshold selection shall be informed by reconstruction experiments in which increasing levels of controlled motion are introduced until reconstruction becomes unreliable or clinically unusable.

Given the practical and ethical constraints of early in-vivo testing, initial validation should be performed on a surrogate setup (e.g., a doll or phantom) that allows repeatable

repositioning and controlled perturbations. The validation should quantify: (i) detection and identification reliability across a representative range of orientations and positions, (ii) repeatability of estimated pose under static conditions, and (iii) sensitivity to small imposed displacements and rotations.

6.1.3 Ultrasound Volume Reconstruction with Motion Correction

If subsequent iterations demonstrate that the patient marker can be tracked with sub-millimetric and sub-degree accuracy, patient motion can be incorporated explicitly into the reconstruction workflow. In that case, the probe pose can be expressed in a patient-fixed reference frame by composing the probe-marker pose and patient-marker pose at each time step. This enables compensation for rigid patient motion during post-processing, potentially allowing reconstruction even when patient movement occurs during the sweep.

However, motion correction is inherently limited by the fact that the patient marker is attached to soft tissue rather than rigid bone. Relative motion between skin and underlying skeletal structures can introduce non-rigid components that cannot be corrected through rigid-body transformations alone. Additional limitations include local soft-tissue deformation due to probe pressure, marker slippage or shear at the skin-foam interface. If motion compensation is implemented, these limitations must be characterized experimentally to determine the conditions under which rigid compensation is valid versus when scan repetition remains necessary.

6.2 Enhance Robustness

The current system represents a first full implementation of an Outside-In tracking pipeline based on HydraMarker detection and identification. While the validation experiments demonstrated that accurate pose estimation is achievable, overall robustness remains strongly dependent on the reliability of the low-level detection stages. In particular, missed corner detections may lead to missing samples within the reconstructed grid, which directly impacts marker identification. This issue is structural: the dot table used for decoding is built from 2×2 blocks of corners, and a dot location is defined only when the four surrounding corners are available, with the dot center computed as their average

[1]. Consequently, corner dropout in the interior of the grid reduces the number of valid dot samples and can result in ambiguous or incorrect alignment during pattern matching [2].

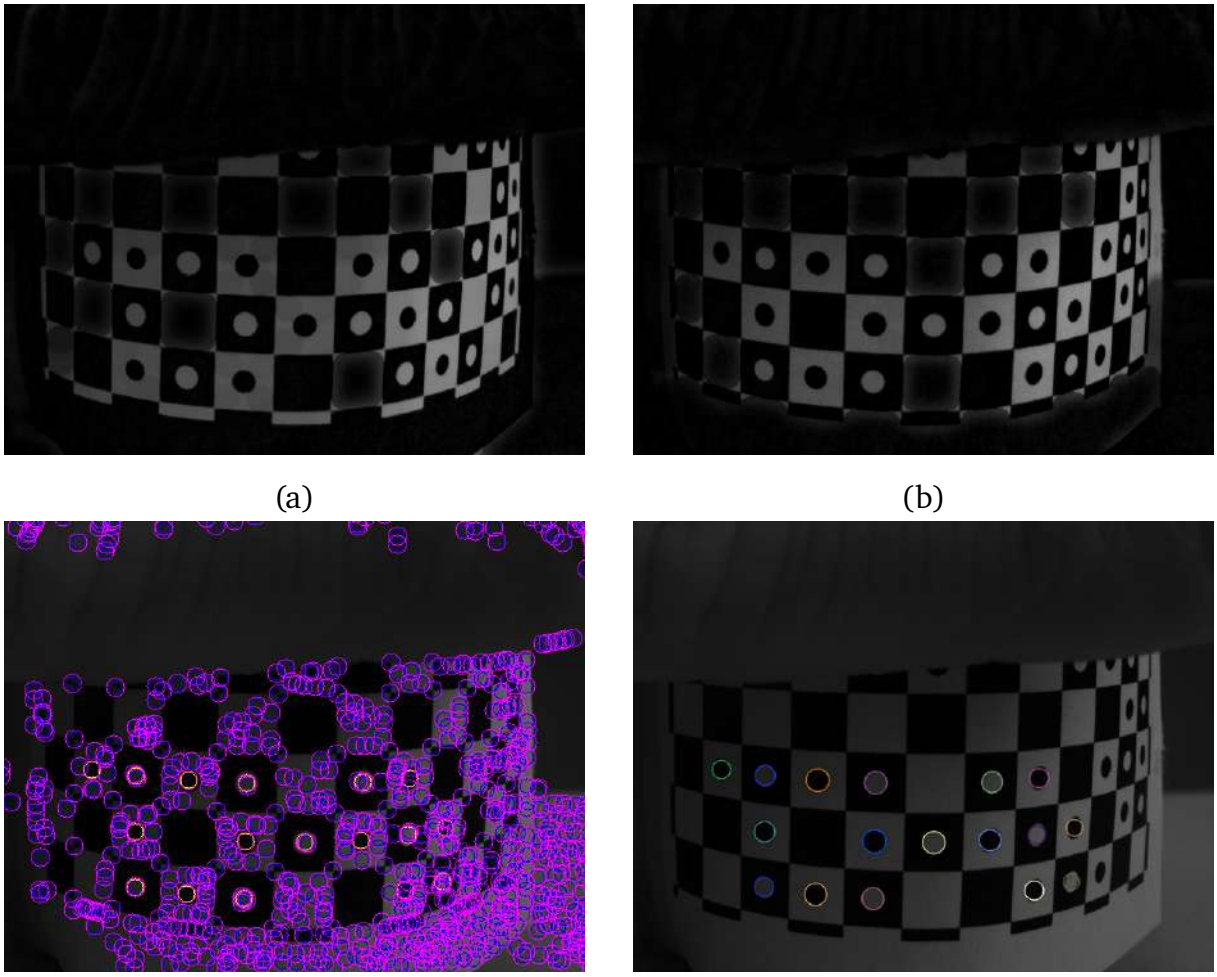
6.2.1 Advanced Corner and Blob Detection Methods

Corner detection in the current pipeline is performed via gradient analysis (Section 3.4.2), and the dominant failure mode observed is the loss of corners within the grid interior. This failure propagates downstream by reducing the number of valid dot samples available for decoding, ultimately increasing the likelihood of mis-identification. A first direction to improve robustness is therefore to evaluate alternative corner detection strategies that are specifically engineered for chessboard-like patterns and are known to be resilient under moderate blur and contrast variations. Without committing to a specific implementation, it is proposed to benchmark additional detectors and refine criteria, including OpenCV's chessboard-oriented detectors as candidate alternatives or fallbacks to the current method.

A complementary direction is to strengthen dot detection, which in the current framework is used exclusively for tag decoding. In the present implementation, dot sampling depends on a correctly formed grid and on the availability of the surrounding corners. For this reason, an alternative dot extraction approach was explored using OpenCV's `SimpleBlobDetector`. A dedicated prototype script was implemented to detect circular dot-like features without requiring a fully reconstructed grid, with the intent of improving dot availability under corner dropout.

The implemented prototype performs a two-pass blob detection strategy on each frame: one pass is optimized for dots that appear dark-on-light and the other for dots that appear light-on-dark. Prior to detection, the image is locally contrast-normalized using CLAHE and enhanced using a combination of morphological operators and background subtraction, producing two enhanced images where dot candidates appear as bright blobs, as can be seen from Figures 6.1 (a–b). The resulting detections from the two passes are merged by spatial proximity to avoid duplicates, and moderate geometric filters are applied to suppress non-dot structures (Figures 6.1 (c–d)). This approach was pursued and implemented, but it was not integrated into the final tracking pipeline, as systematic parameter tuning and quantitative validation were not completed.

[1] See Eq. 3.8 and the description of dot table construction.
[2] See Section 3.4.5



(a) (b)

(c) (d)

Figure 6.1: (a) Pre-processing output for the dark-on-light pass. (b) Pre-processing output for the light-on-dark pass. (c) Blob candidates detected after the two passes and merging. (d) Final subset of blob detections retained as plausible HydraMarker dots for subsequent decoding.

6.2.2 Comprehensive Testing of Unidentified Corner Utilization for Pose Refinement

In addition to improving the primary detection stages, robustness may be enhanced by leveraging information currently discarded by the conservative identification policy. As described in Section 3.6.4, the detector outputs both identified and unidentified corners. Unidentified corners are withheld to avoid introducing incorrect correspondences, but they still encode valid geometric structure and may be exploited to refine pose estimates if matched reliably.

A preliminary refinement stage is already implemented but has not been validated experimentally. Experiments should be conducted to assess when and how this refinement improves performance, and to verify that it does not introduce failure modes under challenging detections (e.g., when the initial pose is biased or when false detections are present).

6.2.3 Kalman Filtering and Optical Flow for Outlier Rejection

The current pipeline includes post-RANSAC validation filters that reject poses deemed physically implausible based on a motion threshold on frame-to-frame translation and a rotation magnitude check on the angle-axis rotation change (see Section 3.6.3.4). While effective at preventing large outliers, these checks rely on hand-tuned thresholds and can be overly strict in scenarios with temporary detection degradation, leading to dropouts.

A more principled approach is to replace (or complement) these hard thresholds with temporal filtering based on a constant-velocity Kalman model. In this framework, the expected pose is predicted from previous frames, and new measurements are accepted or rejected based on a statistically grounded innovation test rather than fixed limits. This enables dynamic gating that adapts to the observed motion regime and measurement noise. Importantly, filtering can be introduced at two levels: (i) at the feature level, to stabilize corner and dot detections prior to identification, and (ii) at the pose level, to smooth the final 6-DoF estimate and reject inconsistent updates.

In addition, optical flow can be used to track features between consecutive frames, providing continuity when re-detection is temporarily unreliable. By propagating corner locations forward in time, optical flow can support re-association of detections, reduce

missed-corner events, and provide stronger initialization for the identification stage. Combined with Kalman-based gating, this approach would reduce reliance on strict heuristic thresholds while maintaining robustness to transient detection failures.

6.3 Real Time Processing

The current system was developed with an offline workflow in mind. In particular, Clarius data are processed post hoc, together with the subsequent reconstruction of the ultrasound volume. For this reason, real-time performance was not a strict design requirement in the present implementation. Nevertheless, achieving real-time tracking remains an attractive direction, primarily to provide immediate feedback to the sonographer during acquisition (e.g., motion flagging and rescan suggestion), even if full 3D reconstruction remains an offline step.

In this context, real-time is defined as an end-to-end processing rate on the order of the camera frame rate (≈ 20 fps), including detection, identification and pose estimation.

6.3.1 C++ Implementation for Faster Computation

A practical path toward real-time performance is to migrate computationally intensive components from Python to compiled code. Although many OpenCV functions are internally optimized, the end-to-end pipeline also includes custom operations and repeated per-frame logic (e.g., corner bookkeeping, dot sampling and decoding, correspondence management), which can introduce non-negligible overhead. Therefore, a first step would be systematic profiling to quantify the time budget of each stage and to identify the dominant contributors.

Based on empirical observations, the corner detection and refinement stage appears to be the principal bottleneck. Its runtime is sensitive to the preprocessing parameters (e.g., Gaussian blur and patch size), which introduce a direct trade-off: stronger smoothing can improve runtime but involves the use of markers with larger features, therefore decreasing the number of usable IDs for pose estimation. A compiled implementation would allow tighter memory control, reduced Python overhead, and more efficient multi-threading.

6.3.2 GPU Acceleration Techniques

Further acceleration may be achieved by exploiting GPU parallelism for pixel-wise and local-neighborhood operations. In practice, the feasibility of GPU acceleration depends on the availability of compatible hardware and software support. In the present project, GPU-based implementations were not pursued due to hardware constraints on the development platform (Surface Book 2: $16\,\mathrm{GB}$ RAM, Intel Core i7-8650U), which limited the ability to prototype and benchmark GPU pipelines.

6.3.3 Combining for Real Time Motion Detection

If end-to-end real-time tracking can be achieved, the primary clinical benefit would be real-time motion detection and immediate feedback during scanning. In a first implementation, this feedback can be intentionally simple: a binary warning indicating that patient motion exceeded an acceptable threshold, coupled with a suggestion to repeat the sweep. This real-time flagging would improve workflow by allowing compromised acquisitions to be identified immediately, without waiting for offline reconstruction and quality assessment.

6.4 Clinical Validation

While bench-top experiments can quantify pose accuracy, repeatability, and robustness under controlled conditions, clinical deployment requires evidence that the Outside-In tracking workflow is feasible in-vivo and that the reconstructed results remain comparable to clinically accepted references, following the Inside-Out validation approach (K. K. Chen 2025). In principle, an intermediate validation step on a large-animal model could be used to study soft-tissue motion and handling in a living subject.

6.4.1 In-Vivo Feasibility Assessment

An in-vivo feasibility phase should be designed to evaluate whether the proposed workflow can be executed reliably on real newborn patients during routine DDH screening, and to identify the main causes of data loss (e.g., occlusion, motion, insufficient visibility of the markers, or incomplete sweeps). Because the Outside-In configuration introduces a second marker on the patient in addition to the probe marker, feasibility should explicitly include

the practical aspects of patient marker placement, temporary fixation (e.g., held by the sonographer or applied with gentle skin-safe adhesive), and visibility under typical hand positions and room lighting. During feasibility, qualitative feedback from sonographers should also be collected (setup time, comfort, interference with typical scanning posture, perceived added workload).

6.4.2 Validation Against 3D Ultrasound

Once feasibility is established, validation against 3DUS should quantify how closely the reconstructed volume obtained from tracked freehand 2D sweeps matches a 3DUS reference acquired from the same hip in the same session. The core purpose is to test whether the proposed system can produce reconstructed anatomy and derived dysplasia metrics that are comparable in value and repeatability to a dedicated 3DUS acquisition.

A practical dataset structure is:

1. Repeated tracked sweeps (Outside-In): two acquisitions per hip (each consisting of multiple sweeps), enabling repeatability analysis under the same session conditions.
2. Reference 3DUS acquisition: one scan per hip, providing a modality baseline for agreement analysis.

Validation should explicitly separate two questions:

- Repeatability: do repeated tracked reconstructions of the same hip agree with each other (OT–OT)?
- Agreement with 3DUS: do tracked reconstructions agree with the corresponding 3DUS reference (3D–OT)?

6.5 Broader Applications of HydraMarker Based Tracking

Given the accuracy results presented in Chapter 4, the HydraMarker based Outside-In tracking pipeline has potential applications beyond the immediate use case of freehand 2D ultrasound tracking for DDH screening. This section outlines several broader application areas where the proposed tracking approach could be beneficial and will be validated by collaborators at the Surgical Technologies Lab (STL) at UBC and at the Centre for Aging SMART.

6.5.1 Objective Knee Osteoarthritis Assessment with Tracked POCUS

A further application of HydraMarker based Outside-In tracking is the development of machine learning methods for objective assessment of knee osteoarthritis using point-of-care ultrasound (POCUS) devices. The goal is to enable a standardized and reproducible approach to imaging knee cartilage and osteoarthritis features with handheld ultrasound systems. In this application, the ultrasound device is a Clarius HD3 L15, which has the same probe housing dimensions as the Clarius HD3 L7 used in the DDH experiments. Consequently, the probe-mounted HydraMarker integration and the Outside-In tracking workflow can be transferred directly, requiring only minor mechanical adjustments if a different camera is adopted.

A key challenge in current clinical and research practice is the subjectivity of ultrasound image acquisition. Variability in probe placement and orientation introduces both intra- and inter-rater differences, which propagate into downstream measurements. This is particularly critical for cartilage thickness estimation, where small changes in probe tilt or out-of-plane alignment can produce substantial differences in the measured thickness, even when the resulting image appears visually acceptable.

The proposed tracking pipeline provides a practical mechanism to reduce acquisition variability by shifting the workflow from single-frame imaging toward tracked sweep acquisition. Rather than relying on a single operator-selected “best” image, a short sweep can be acquired while tracking probe pose, enabling retrospective selection of an optimal frame according to a standardized criterion. A machine learning model can then be trained to identify the optimal frame(s) from the sweep, followed by dedicated segmentation models to extract the relevant anatomy and compute objective measurements.

In future work, the same tracked acquisition could be extended toward volumetric reconstruction of the knee cartilage which would enable more comprehensive and spatially consistent measurements than single-plane thickness estimates.

6.5.2 Intraoperative Surgical Tools for Advanced Reconstruction Project

In collaboration with members of the ISTAR group within the STL, HydraMarker based tracking will be evaluated as a solution for tracking bone clamps used during mandibular

reconstruction with fibular free flap procedures. The current clinical and research standard commonly relies on an NDI Vega optical tracker with passive spherical marker tools attached to the clamp. Although reliable, these tools are relatively bulky, can be partially occluded in the operative field, and may suffer from tracking dropouts during critical steps of the workflow.

HydraMarker based tracking may offer a more compact and cost-effective alternative while maintaining accurate pose estimation, particularly when the marker layout can be tailored to the available mounting surfaces on the clamp. To evaluate feasibility, validation experiments will be conducted using multiple HydraMarker sizes (ranging from $20\,\mathrm{mm} \times 20\,\mathrm{mm}$ to $40\,\mathrm{mm} \times 40\,\mathrm{mm}$) and a Basler daA2448-70uc camera. In this application, multi-target tracking is essential, as three clamps and a reference object are expected to be visible within the same camera frame. The evaluation will therefore focus not only on single marker accuracy, but also on simultaneous detection, identification, and pose estimation of multiple markers.

6.5.3 Depth Camera Augmented Fluoroscopy Project

The HydraMarker tracking pipeline is also directly relevant to ongoing work within the DeCAF project. The latest development stage of DeCAF focuses on achieving an accurate overlay of an intraoperative X-ray image onto the camera view. While this overlay represents an important milestone, recent clinical feedback identified a key limitation of the current prototype: the lack of surgical tool guidance. In particular, the need for drill guidance was emphasized, including visualization of the drill axis and the ability to estimate distance information (e.g., insertion depth) during drilling.

The tracking approach developed in this thesis provides a practical foundation to address these requirements. By enabling reliable Outside-In pose estimation of a surgical instrument equipped with a compact fiducial marker, the system can provide the instrument pose relative to the camera coordinate frame. This information can be integrated with the existing X-ray overlay pipeline to support augmented guidance. In addition, standard calibration procedures such as pivot calibration can be employed to estimate the tool tip position relative to the tracked marker frame, enabling direct visualization of both the tool tip and the drill axis in the camera image and within the X-ray overlay.

Once the tool tip and axis are available, the drilling progress can be quantified

by projecting the tracked axis into the overlay and estimating insertion depth along the axis direction. From a clinical perspective, this enables more precise guidance, supports intraoperative awareness of drill trajectory and depth, and may reduce the risk of unintended penetration into anatomically critical regions. Future work within DeCAF will therefore focus on integrating HydraMarker based instrument tracking with the overlay pipeline and evaluating guidance accuracy under representative procedural conditions.

Chapter 7

Discussion and Conclusions

7.1 Discussion

The objective of this work was to quantify the performance of a single-camera, Outside-In HydraMarker-based tracking system for localizing a 2D ultrasound probe at clinically relevant working distances. Overall, the validation results in Chapter 4 indicate that the proposed setup achieves sub-degree rotational accuracy and sub-millimetric translational accuracy, with high repeatability in both rotation and translation, while maintaining reliable marker identification under severe occlusion scenarios. In addition, multi-marker experiments demonstrate that multiple HydraMarkers can be detected, correctly identified, and tracked simultaneously within the same camera field of view, without evidence of mislabelling. However, pose accuracy under severe occlusion and under multi-marker conditions was not quantified in this study. Taken together, these results suggest that the proposed tracker provides accuracy and repeatability consistent with the requirements of tracked 2D ultrasound scanning for neonatal hip assessment.

At the selected working distance, the tracking algorithm exhibits high repeatability and stable accuracy. The system reliably tracks large angular ranges, reaching approximately 140° about the transverse axis, 110° about the beam axis, and 360° about the elevation axis. The larger incremental range observed about the transverse axis is attributed to the probe-marker geometry: during the incremental motion the probe tilts towards the camera, and tracking can be maintained until only a minimal portion of the pattern remains visible (approximately one marker row). Conversely, during the decremental rotation the marker becomes highly foreshortened and the checker edges shorten, which

likely reduces the number of usable corners (and causes corner clustering). This limitation is not considered critical given the large usable range achieved. The asymmetric range about the beam axis (Fig. 5.14) is plausibly related to the axis placement being offset relative to the marker center. Further testing to isolate this effect was not performed, as the achieved ranges and errors were already satisfactory. Across 360° elevation rotations, no abrupt pose flips or ID swaps were observed, suggesting robust marker identification even under strong tilt and full rotations. Finally, it is worth noting that the tracker behaves conservatively: when insufficient visual information is available it stops tracking instead of producing erroneous estimates.

Rotational performance was assessed using an end-point metric derived from temporally averaged poses at the beginning and end of each trial. To reduce sensor noise, a temporal average over a fixed window was applied at the start and end of each trajectory (static start and static end), and the end-point rotation difference was then computed. This protocol provides a precise estimate of the net angular discrepancy, but it is not directly comparable to studies that compute point-wise discrepancies along the full trajectory and derive per-trajectory RMS values, such as the evaluation of an Inside-Out tracked 2D ultrasound system against an NDI Vega ST optical tracker in Piwowarczyk et al. (2025). Across all axes and both equally represented configurations, the combined end-point rotational RMSE was 0.60° (mean $= -0.56^\circ$, SD $= 0.21^\circ$, MAE $= 0.56^\circ$, $N = 58$). Since RMSE includes systematic bias, a complementary bias-removed precision metric was also computed by subtracting the mean error per configuration before pooling, yielding a precision RMSE of 0.09°. In addition, pooling all axes and both incremental rotation directions, the combined RMSE across all increments was 0.31° ($N = 118$). The lower RMSE observed in the incremental tests compared with the end-point evaluation is primarily explained by the role of systematic bias in the end-point metric: in the pooled end-point results the mean error is -0.56°, indicating that RMSE is dominated by a consistent offset rather than trial-to-trial variability. This interpretation is supported by the bias-removed precision estimate (0.09°), which indicates high repeatability once configuration-dependent offsets are removed.

These results can be contextualized by comparison with previous work. Piwowarczyk et al. (2025) evaluated an Inside-Out tracking approach in which a camera is mounted on the ultrasound probe and observes an external 3D ArUco marker array, using an NDI

Vega ST as the ground-truth reference. In that study, rotational accuracy was reported as a trajectory-based RMSE with mean offset removed, yielding a rotational RMSE of approximately 0.37° over clinically relevant trajectories. This protocol is conceptually closer to the incremental analysis used here than to the end-point metric. The combined incremental RMSE of 0.31° for the present Outside-In system therefore falls within the same range as the Inside-Out setup.

For each axis, translational performance was quantified by computing the incremental RMSE over all targets in the combined set, yielding axis-wise RMSE values reported in Tables 5.10–5.12. To obtain a single overall incremental translation accuracy metric across axes, a weighted pooled RMSE was computed from the axis-wise RMSE values (weighted by the number of samples per axis), resulting in a pooled incremental translational RMSE of $0.09\,\mathrm{mm}$ ($N = 49$). For comparison, Piwowarczyk et al. (2025) report a translational discrepancy RMSE of $0.31\,\mathrm{mm}$ for their Inside-Out system. Although the protocols differ (incremental displacements with caliper ground truth versus continuous trajectories with optical tracking ground truth), these values indicate that the proposed Outside-In system achieves translational accuracy that is well within the range of existing systems based on Inside-Out tracking.

Additional context is provided by the position-sensing marker system described by Xie et al. (2023). In that work, a binocular camera pair and a dense multi-fold HydraMarker pattern are used to track surgical instruments and a pelvic phantom during minimally invasive total hip arthroplasty. Positioning accuracy is evaluated by comparing tracked displacements of a probe tip against a Vernier caliper over $50\,\mathrm{mm}$ to $150\,\mathrm{mm}$ motions in depth and baseline directions, yielding an overall mean displacement error of $0.33\,\mathrm{mm}$ with a standard deviation of $0.18\,\mathrm{mm}$. In the present study, the mean error magnitude at an $80\,\mathrm{mm}$ displacement was $0.17\,\mathrm{mm}$ in the baseline (transverse) direction (signed mean: $0.17\,\mathrm{mm}$) and $0.33\,\mathrm{mm}$ in the depth (elevation) direction (signed mean: $-0.33\,\mathrm{mm}$). Using linear interpolation of the Xie et al. (2023) mean errors reported for $50\,\mathrm{mm}$ and $100\,\mathrm{mm}$ displacements, the corresponding values for $80\,\mathrm{mm}$ are approximately $0.19\,\mathrm{mm}$ (baseline) and $0.35\,\mathrm{mm}$ (depth), which are comparable to the magnitudes observed here, despite substantial differences in tracking architecture (single-camera versus binocular stereo), evaluation protocol, tracked objects, and working distances (approximately $300\,\mathrm{mm}$ here versus $1500\,\mathrm{mm}$ in the MITHA navigation system).

The distribution of translational errors in the present experiments is anisotropic: the errors of the elevation axis are larger than those of the transverse axis. This behaviour is consistent with the expected depth estimation challenges of monocular pose estimation. In a single-camera setup, the geometric constraints provided by perspective projection are fundamentally anisotropic: lateral motion within the image plane generates large changes in pixel coordinates, whereas motion along the camera's optical axis (depth) produces only modest perspective scaling that depends on the inverse of distance. Consequently, depth estimates derived from PnP algorithms rely heavily on subtle variations in the projected marker size and corner positions, making them more sensitive to pixel-level localization noise, lens distortion residuals, and calibration uncertainties than estimates of lateral displacement. For stereo or binocular systems, depth uncertainty is known to scale approximately as $\sigma_Z \propto Z^2/(b \cdot f)$, where Z is the range (distance from the camera(s) to the target object), b is the stereo baseline, and f is the focal length, reflecting the ill-conditioning of triangulation as the intersection angle between viewing rays decreases. In monocular pose estimation, the absence of a physical baseline further degrades depth observability, since all geometric constraints derive from a single viewpoint. Nonetheless, being located at a relatively short working distance, approximately one fifth of the maximum range evaluated by Xie et al. (2023), the present setup achieves depth (elevation) errors that remain within a clinically acceptable sub-millimetric range and comparable to those reported for stereo-based HydraMarker navigation systems.

When comparing these results to clinical-grade Outside-In optical trackers such as the NDI Vega XT, manufacturer documentation reports translational RMSE on the order of $0.12\,\mathrm{mm}$ for marker localization and rotational RMSE of approximately $0.05°$ under optimal conditions within the specified working volume (NDI 2024). Within this context, the proposed system attains translational accuracy that is comparable in magnitude to that of Vega, albeit over a more limited working volume and under different imaging conditions, while exhibiting larger rotational errors (incremental RMSE $0.31°$). This discrepancy in rotational performance is expected, given that the proposed setup uses a single RGB camera, vision based fiducials, and a general-purpose PnP pipeline, whereas Vega employs multiple infrared cameras, retroreflective spheres, and proprietary calibration and reconstruction algorithms optimized for surgical navigation. Nevertheless, the achieved accuracy appears sufficient for the intended application of reconstructing 2D

ultrasound sweeps for neonatal hip assessment.

Finally, it is important to acknowledge the influence of the evaluation protocol on the reported variability metrics. The standard deviations of the tracking errors are calculated after frame averaging over a fixed temporal window at the start and end of each trial, which reduces the influence of transient noise and short-term jitter. This approach is appropriate for quantifying end-point discrepancies and for isolating systematic biases, but it may not fully capture the variability observed in continuous real-time tracking. Therefore, while the reported SD values indicate low dispersion under the specific, temporally averaged conditions used in this study, they likely underestimate the instantaneous frame-to-frame variability that would be encountered during online tracking without averaging. Future work should therefore complement the present static and quasi-static evaluations with dynamic trajectories and real-time analysis to more fully characterize temporal noise and its impact on ultrasound volume reconstruction.

7.2 Limitations of Current Implementation

7.2.1 Limited Angular Validation of Jitter and Translational Accuracy

When analyzing the jitter, the current validation was performed with the camera positioned approximately perpendicular to the marker, and the probe held in a single representative orientation. This configuration is idealized and does not fully reflect the range of viewing angles encountered in clinical neonatal hip scanning. As a result, the jitter metrics reported in this work characterize temporal pose stability only for this perpendicular configuration, and the behaviour of the system under more oblique viewing angles remains unknown, even though it is expected to be comparable.

Similarly, translational accuracy was assessed using a setup in which the camera was aligned frontally to the marker and motion was applied along axes that are approximately aligned with the camera's line of sight. While this configuration is useful to establish a lower bound on translational error, it does not capture potential degradation in accuracy that may arise when the probe is viewed at different incidence angles. In realistic clinical scenarios, strict perpendicular alignment between camera and probe may not be achievable, and the sonographer's ergonomics should take precedence over camera positioning. Therefore, the consistency of both jitter and translational error across multiple

viewing angles has not yet been validated and should be investigated before proceeding to in-vivo clinical studies.

7.2.2 ID Tag Size Selection and Marker Field Size Trade-off

The identification robustness of the marker tracking system depends critically on selecting an appropriate ID tag size for the geometry being tracked. This selection is constrained by the properties of the fast-bWFC marker field generation algorithm and the visibility characteristics of the target object.

Smaller ID tags enable the sampling of corners under severe occlusion, as fewer cells need to be simultaneously visible to correctly identify a marker region. However, the fast-bWFC algorithm imposes computational and combinatorial constraints: smaller ID tags result in fewer unique possible patterns, which limits the maximum size of the marker field that can be generated while maintaining the required uniqueness constraints. Conversely, larger ID tags provide access to significantly more unique ID patterns, allowing substantially larger marker fields to be generated, but at the cost of requiring more cells to be visible simultaneously for correct identification. An example of this limitation is shown in Figure 7.1, where an inappropriately large ID tag size results in insufficient visible cells for correct identification. The correct identification and numbering is shown in Figure 5.33 (a–f).

The user must carefully select an ID tag size that balances the visibility constraints of their target geometry against the marker field size requirements of their application. The confidence threshold described in Section 5.3 partly solves this limitation by rejecting frames where less than 15% of the marker is visible and prevents wrong identification. If the correct ID tag size is selected upfront, the confidence threshold may be relaxed or removed, as corners should be correctly identified even under severe occlusion.

Figure 7.1: Misidentification due to inappropriate ID tag size

7.2.3 Marker Field Uniqueness Across Multiple Instances

When generating multiple distinct marker field instances using the fast-bWFC algorithm, the current implementation does not guarantee that all generated fields are completely different from one another. This limitation was empirically observed: generating the marker field twice without perturbation seeding produced identical results, confirming that the algorithm is deterministic and produces the same output given the same initial conditions.

To mitigate this issue, the generation process employs a perturbation strategy: 30% of cells are preset during the initialization phase before the fast-bWFC fill proceeds. This ensures that different random seeds lead to different marker field instances through initial diversity, though no explicit verification is performed to guarantee complete distinctness across all generated fields.

For the current application requiring only two distinct marker fields, this probabilistic approach proved sufficient. The perturbation strategy successfully generated two visually and functionally distinct markers that were correctly identified and disambiguated during

tracking experiments, as described in Section 5.4. However, for applications requiring numerous independent marker fields, a more rigorous approach such as explicit post-generation Hamming distance comparison would be prudent to ensure all fields are guaranteed to be unique.

7.2.4 Marker Occlusion and Overlap Limitations

A notable limitation arises when two marker fields are positioned in depth-aligned configurations where one occludes parts of the other in the camera view. As illustrated in Figure 7.2, when markers overlap in projection, the corner detection algorithm fails to distinguish the boundary between the two distinct marker fields. Instead, it treats the visible portions as a single continuous marker surface, merging corner detections across both markers. This causes the identification algorithm to sample an inconsistent mixture of ID patterns from both markers simultaneously, resulting in erroneous identifications.

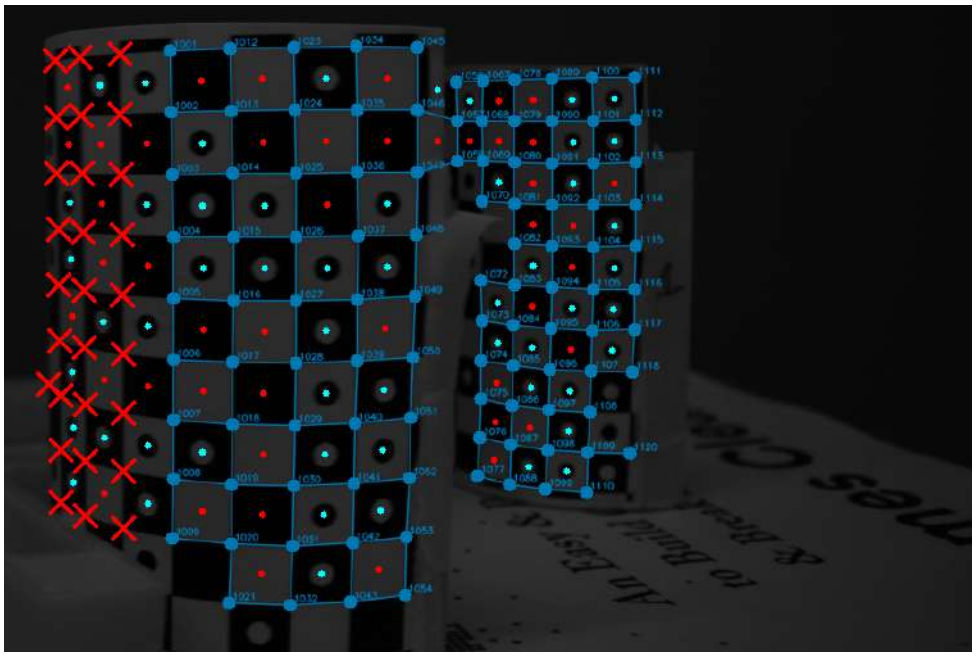


Figure 7.2: Depth-aligned marker overlap with merged corner detection across both markers and erroneous identifications

7.2.5 Post-Processing Computational Constraints

The current implementation of the tracking pipeline is entirely offline. After an initial calibration step, ultrasound acquisitions are recorded while the camera streams images to disk, and all subsequent processing steps (marker detection, identification, and pose

estimation) are executed in post-processing. This design greatly simplifies experimental validation and algorithm development, but it introduces a practical limitation for clinical deployment: no tracking feedback is available to the sonographer during the scan, and motion related artefacts or tracking failures can only be detected retrospectively. As a consequence, corrupted acquisitions may need to be repeated, and real-time guidance or motion warning functionality cannot yet be provided at the point of care.

With the final marker design and the chosen Gaussian smoothing parameters, the optimized implementation achieved detection and identification for 10 frames in approximately $1\,\mathrm{s}$, while the camera recorded at $20\,\mathrm{Hz}$. Solving PnP for these 10 frames required an additional $0.2\,\mathrm{s}$, yielding a total processing time of about $2.4\,\mathrm{s}$ per $1\,\mathrm{s}$ of video. This $\sim 2.4\times$ slowdown relative to real time confirms that further optimization of the detection and identification stages (e.g., C++ reimplementation and/or GPU acceleration) will be necessary before the system can support interactive use in the clinic.

7.2.6 Optimization Algorithm

The marker design optimization framework presented in this work was introduced primarily as a structured starting point for design selection, rather than as a mathematically rigorous global optimizer. Its main contribution is to replace ad-hoc parameter guessing with a reproducible procedure that filters infeasible designs via explicit constraints and ranks the remaining candidates according to simple objective criteria.

A first limitation is the formulation of the objective functions. The *maximum identification capacity* and *maximum robustness* objectives each collapse the design decision to a single scalar criterion with heuristic tie-breaking. While this is convenient, it does not explicitly encode how much robustness should be traded for identification capacity (or vice versa) for a given application. The *balanced compromise* objective partially addresses this by using a compromise-programming approach, selecting the candidate with minimum Euclidean distance to a utopia point in a normalized two-objective space. However, this formulation implicitly assumes equal importance of the two objectives and equal sensitivity along both axes. In addition, the normalization depends on the minimum and maximum values observed in the feasible candidate set, making the resulting ranking sensitive to the boundaries of the explored design space and to the constraint set used to define feasibility. As a consequence, the selected "best" compromise can shift if the

candidate space is expanded, if constraints are modified, or if objective distributions are skewed by outliers.

A second limitation is the search strategy. The current implementation performs an exhaustive parameter sweep over discrete values of the Gaussian smoothing parameter and the corner-refinement patch size. Although this brute-force approach is tractable for the limited two-parameter search used here, it does not exploit any structure in the objective landscape and does not generalize efficiently as additional design variables are introduced (e.g., tag size, field dimensions, shape constraints). More rigorous approaches could reduce the reliance on discrete sampling and provide stronger guarantees of convergence to high-quality solutions. These extensions would preserve the original motivation of the optimization step while improving robustness of the selection process and scalability to richer design spaces.

7.2.7 Lack of In-Vivo Clinical Validation and Lighting Specifics

A further limitation of the present work is that the tracking pipeline was validated exclusively through bench-top laboratory experiments. Although these controlled tests are appropriate to quantify pose accuracy, jitter, identification robustness, and failure modes under repeatable conditions, they do not fully represent the clinical environment of neonatal hip scanning. In the clinic, additional sources of variability are expected, including patient motion, operator-dependent handling, a wider range of viewing angles, variable illumination, time constraints, and practical workflow factors related to marker placement and occlusion by the operator's hands.

Moreover, all experiments were conducted under ambient laboratory lighting without reproducing the illumination conditions of a clinical examination room. In practice, neonatal hip scans are often performed in dimmed environments, and additional equipment or personnel can introduce shadows or glare. Although the tracking algorithm demonstrated robust performance without a dedicated illumination source in this study, the impact of clinical lighting conditions on detection reliability and pose accuracy remains unquantified.

Consequently, the results reported in this thesis should be interpreted as an initial technical validation. Before translation to clinical use, the system must be evaluated in vivo to assess usability and acquisition success rates, to characterize tracking performance

under realistic scanning conditions, and to determine whether the reconstructed outputs remain clinically reliable. These aspects are addressed as future work in Section 6.4.

7.2.8 Lack of a Clinician-Facing Interface and Workflow Integration

A practical limitation of the current implementation is that the full tracking workflow is not yet usable by sonographers as a standalone tool. The system currently requires running scripts, managing file paths, and executing calibration, acquisition, and post-processing steps manually. No dedicated graphical user interface (GUI) is provided to guide the operator through the workflow, enforce correct procedure, or present quality feedback during acquisition.

As a consequence, the present implementation is best suited for research use by technical users rather than routine clinical operation. For translation to clinical studies, a clinician-facing interface should be developed to streamline the acquisition pipeline, automate calibration and data handling, and provide clear prompts and status indicators. In addition, basic runtime checks (e.g., marker visibility and motion warnings) should be exposed to the user to reduce the likelihood of unusable acquisitions and to support consistent use across operators. Although the current Python-based implementation does not yet achieve real-time performance, the measured processing throughput suggests that a carefully optimized version could feasibly operate in real time at clinically relevant frame rates.

7.3 Contributions and Conclusions

7.3.1 Outside-In Tracking Architecture for Ultrasound Probes

A single-camera, Outside-In optical tracking architecture is presented for localizing a 2D ultrasound probe during neonatal hip scanning. Deformable, multifold locally unique HydraMarkers are mounted on the probe and, in a proposed extension, on the patient support to enable both robust probe tracking and a pathway to patient-motion observability. Unlike prior Inside-Out implementations this configuration combines probe ergonomics and potential patient-motion awareness, while retaining sub-millimetric and sub-degree accuracy, as quantified in Chapter 4.

7.3.2 Marker Design Optimization and Custom HydraMarker Layouts

A design optimization framework is formulated to select HydraMarker field dimensions, cell sizes, and detector parameters under optical, printing, and workflow constraints. By combining a projection-based visibility model with printability and minimum-area limits, the method generates feasible marker candidates and ranks them via multi-objective criteria. This leads to custom HydraMarker layouts tailored to tool geometries, balancing visibility under realistic scenarios and the number of usable IDs for pose estimation.

7.3.3 End-to-End Detection, 3D Reconstruction, and Pose Estimation

An end-to-end image-based pipeline is implemented to convert raw camera frames into six-degree-of-freedom probe pose for HydraMarker fields. The pipeline includes: (i) a multi-stage corner detector and identifier, (ii) a structure-from-motion reconstruction with bundle adjustment to produce a metric 3D marker map, and (iii) robust pose recovery via PnP with RANSAC outlier rejection. A complete calibration chain is also established, registering the camera frame to the ultrasound image plane through optical-tracker-based registration in 3D Slicer, enabling localization of each ultrasound frame in the camera coordinate system.

7.3.4 Validation Protocol and Quantitative Characterization

A dedicated validation framework is developed using precision rotational and translational stages as ground truth. The protocol defines jitter metrics, end-point and incremental accuracy measures, repeatability tests, and robustness experiments under extreme occlusion and multi-marker conditions. With the final 6×10 HydraMarker configuration, the system achieves approximately 0.01° rotational jitter and $0.005\,\mathrm{mm}$ translational jitter, with sub-degree rotational RMSE and sub-millimetric translational RMSE. Reliable identification is retained under more than 90% occlusion, and multiple markers can be disambiguated within a single camera view without observed mislabelling.

7.4 Conclusions

This work demonstrates that a compact, single-camera, Outside-In HydraMarker-based tracker can localize a 2D ultrasound probe with sub-degree rotational accuracy, sub-millimetric translational accuracy, and high repeatability at clinically relevant working distances. Robust identification under severe occlusion and consistent multi-marker disambiguation support the feasibility of extending the architecture toward patient-motion-aware tracked ultrasound for neonatal hip assessment. The presented system and evaluation protocol provide a technical foundation, validated in a laboratory setting, for translation to real-time operation and subsequent in-vivo feasibility studies.

Appendix A

Marker Design Optimization: Mathematical Formulation and Implementation

This appendix provides detailed mathematical formulations, design constraints, objective functions, and the optimization algorithm employed to systematically select marker parameters for the ultrasound probe. The complete implementation is available in the accompanying Python module `OptimizeMarkerSize.py` and is publicly released on GitHub at https://github.com/samuv01.

A.1 Design Constraints

Several hard constraints bound the feasible design space. All constraints must be satisfied simultaneously; candidates violating any constraint are filtered from consideration.

A.1.1 Physical Marker Dimensions

The marker must fit within the physical dimensions of the carrier object. For the ultrasound probe marker:

$$W_{\text{probe}} \leq 130 \text{ mm}, \quad H_{\text{probe}} \leq 120 \text{ mm} \tag{A.1}$$

For the patient marker support:

$$W_{\text{patient}} \leq 70, \quad H_{\text{patient}} \leq 60 \tag{A.2}$$

A.1.2 Image Area Constraint

The marker's projection onto the camera sensor must not exceed a maximum fraction of the image area. This preserves the field of view and allows the sonographer a greater range of motion during ultrasound acquisition. Using the thin-lens orthogonal projection model, the projected area (in pixels) is:

$$A_{\text{proj,px}} = \left(\frac{f \cdot W_{\text{marker}}}{D_{\text{working}} \cdot p}\right) \times \left(\frac{f \cdot H_{\text{marker}}}{D_{\text{working}} \cdot p}\right) \tag{A.3}$$

where:

- $f = 4$ mm is the focal length of the Edmund Optics lens
- D_{working} is the working distance (mm)
- $p = 0.00345$ mm/pixel is the pixel size of the Basler daA1920-160uc sensor
- $W_{\text{marker}}, H_{\text{marker}}$ are the physical marker dimensions

The constraint is:

$$\frac{A_{\text{proj,px}}}{A_{\text{image,px}}} \leq F_{\max} \tag{A.4}$$

where $A_{\text{image,px}} = 1936 \times 1216 = 2,355,776$ pixels is the total sensor resolution and $F_{\max}$ is the maximum allowable fraction (typically 0.25, but adjustable based on workflow). This constraint is evaluated at the closest working distance $D_{\min} = 250$ mm, as this represents the worst-case (largest) projection.

A.1.3 Printer Resolution Constraint

All printed features must exceed the minimum feature size achievable by the fabrication device:

$$\text{feature\_size} \geq 0.06 \text{ mm} \tag{A.5}$$

This limit is determined by the Xerox Altalink C8055 printer at 1200 DPI. Features smaller than this threshold will not print reliably and corners will be lost due to ink spreading or dropout. This constraint is checked during the parameter sweep; any (σ, r) pair that would require features smaller than 0.06 mm is skipped.

A.1.4 Minimum Identification Count

The marker design must support a minimum number of unique identifiable corners:

$$N_{\text{IDs}} \geq N_{\min} \tag{A.6}$$

This threshold provides adequate redundancy for robust tracking even if some corners are temporarily occluded or poorly detected. For the initial probe marker optimization, $N_{\min} = 40$ was used to ensure a lower bound comparable to the number of corner features in the ChArUco board employed in the Inside-Out reference approach (approximately 45–50 features).

A.1.5 Patch Coverage Proportionality Guard

A critical practical constraint ensures the detection patch size is proportionate to the projected cell size at the closest working distance. If the patch is too large relative to the cell, the local neighborhood sampled during corner detection becomes dominated by background, reducing the ability to discriminate true corners from noise.

The constraint requires:

$$(2r + 1) \geq f_{\text{patch}} \times w_{\text{cell,px}}(D_{\min}) \tag{A.7}$$

where $f_{\text{patch}} = 0.105$ is the minimum coverage fraction, $w_{\text{cell,px}}(D_{\min})$ is the projected cell width in pixels at the closest distance, and $(2r + 1)$ is the patch size. Rearranging:

$$2r + 1 \geq 0.105 \times \frac{f \cdot \text{cell\_size\_mm}}{D_{\min} \cdot p} \tag{A.8}$$

Configurations failing this guard are skipped during the search, ensuring the detection window is appropriately scaled to the feature size.

A.2 Design Variables

The optimization search spans the following tunable parameters:

A.2.1 Gaussian Smoothing Parameter

The corner detection algorithm applies Gaussian smoothing with standard deviation σ before computing image gradients. The Gaussian kernel size is:

$$\text{kernel\_size} = 2\lceil 3\sigma \rceil + 1 \tag{A.9}$$

Higher σ increases the kernel, causing stronger smoothing that suppresses small features and noise. The minimum detectable feature size (in pixels) is determined by the larger of the kernel size and the detection patch size. For the probe marker optimization, the search range was $\sigma \in [0.6, 3.0]$ in steps of $\Delta\sigma = 0.1$.

A.2.2 Corner Detection Patch Parameter

The corner detection patch is a square window of size:

$$\text{patch\_size} = 2r + 1 \tag{A.10}$$

This patch defines the local neighborhood over which corner strength (Harris corner response or similar metric) is computed. Larger patches require more locally consistent edge structures, effectively filtering out small or isolated features. The search range is $r \in [1, 12]$, yielding patch sizes from 3×3 to 25×25 pixels.

A.2.3 Grid Geometry

The marker can employ square grids (equal spacing in x and y) or rectangular grids (independent spacing in each direction). For a square $N \times N$ grid, the number of corners is:

$$N_{\text{IDs,square}} = (N + 1)^2 \tag{A.11}$$

where $N = \min(G_x, G_y)$ is the number of cells in each direction. For a rectangular $G_x \times G_y$ grid:

$$N_{\text{IDs,rect}} = (G_x + 1)(G_y + 1) \tag{A.12}$$

The optimization evaluates both geometries for each (σ, r) pair and selects the one yielding superior results under the chosen objective function.

A.3 Optical and Detectability Model

The relationship between physical feature size and projected size in pixels follows the pinhole camera projection model. Given a required minimum feature size in pixels $s_{\text{min,pixels}}$ (determined by the detection algorithm), the corresponding physical feature size is:

$$S_{\text{object}} = \frac{D_{\text{working}} \cdot s_{\text{min,pixels}} \cdot p}{f} \tag{A.13}$$

where:

- $s_{\text{min,pixels}}$ = required feature size in pixels for robust detection
- S_{object} = physical feature size (mm)
- $f = 4$ mm (focal length)
- D_{working} = working distance (mm)
- $p = 0.00345$ mm/pixel (pixel size)

For a given (σ, r) pair, the minimum feature size in pixels required for robust detection is:

$$s_{\text{min,pixels}} = \max\left(\text{kernel\_size}(\sigma), \text{patch\_size}(r)\right) \tag{A.14}$$

This ensures that the physical corner is large enough to survive Gaussian smoothing and occupies sufficient area within the detection patch. Using the optical projection model, this translates to a required physical cell size at the farthest working distance:

$$\text{feature\_size\_mm} = \frac{s_{\text{min,pixels}} \cdot p \cdot D_{\max}}{f} = \frac{s_{\text{min,pixels}} \cdot 0.00345 \cdot 350}{4} \tag{A.15}$$

Computing at the farthest distance $D_{\max} = 350$ mm ensures the feature remains detectable across the entire working range $[250 \text{ mm}, 350 \text{ mm}]$.

Given the required physical feature size, the maximum number of whole cells (or grid spacing units) that fit in each direction is:

$$G_x = \left\lfloor \frac{W_{\text{marker}}}{\text{feature\_size\_mm}} \right\rfloor, \quad G_y = \left\lfloor \frac{H_{\text{marker}}}{\text{feature\_size\_mm}} \right\rfloor \tag{A.16}$$

For a square grid, $N = \min(G_x, G_y)$, yielding $N_{\text{IDs}} = (N+1)^2$ corners. For rectangular, $N_{\text{IDs}} = (G_x + 1)(G_y + 1)$.

A.4 Objective Functions

Three optimization objectives are explored, each representing a different design philosophy. All objectives respect the constraints listed in Section A.1.

A.4.1 Maximum Identification Capacity

This objective prioritizes the number of unique identifiable corners, which improves tracking robustness through redundancy:

$$\max \quad N_{\text{IDs}} \tag{A.17}$$

Ties are broken by preferring larger $s_{\text{min,pixels}}$, which implicitly increases robustness. This strategy attempts to pack as many detectable features as possible into the available marker area, resulting in higher grid density and more corners per unit area.

A.4.2 Maximum Robustness

This objective prioritizes the largest minimum feature size, ensuring high detection reliability even under challenging conditions (poor contrast, motion blur, lighting variation):

$$\max \quad s_{\text{min,pixels}} \tag{A.18}$$

Ties are broken by preferring higher N_{IDs}. This typically yields fewer, larger cells with more reliable corner detection at the cost of reduced identification capacity.

A.4.3 Balanced Compromise

This objective seeks a middle ground between capacity and robustness using L^2 compromise programming (Euclidean distance to a utopia point). Both objectives are first normalized to the range $[0, 1]$:

$$N_{\text{IDs,norm}} = \frac{N_{\text{IDs,candidate}} - N_{\text{IDs,min}}}{N_{\text{IDs,max}} - N_{\text{IDs,min}} + \epsilon} \tag{A.19}$$

$$s_{\text{norm}} = \frac{s_{\text{min,pixels,candidate}} - s_{\text{min,pixels,min}}}{s_{\text{min,pixels,max}} - s_{\text{min,pixels,min}} + \epsilon} \tag{A.20}$$

where $\epsilon = 10^{-12}$ is a small constant to prevent division by zero. The utopia point (theoretically best outcome for both objectives simultaneously) is $(1, 1)$. For each candidate, the Euclidean distance to utopia is computed:

$$d_{\text{utopia}} = \sqrt{(1 - N_{\text{IDs,norm}})^2 + (1 - s_{\text{norm}})^2} \tag{A.21}$$

The candidate with the smallest distance is selected as the balanced solution. Intuitively, this metric favors designs that excel in both objectives rather than extremes that maximize only one at the expense of the other.

A.5 Optimization Algorithm: Probe Marker

The algorithm performs an exhaustive sweep over the discrete parameter space for the ultrasound probe marker. Pseudocode is provided below.

A.5.1 Pseudocode: Exhaustive Parameter Sweep

```
FUNCTION optimize_probe_marker(params):
    // Pre-compute global image-area constraint (same for all candidates)
    D_min = 250 mm  // closest working distance
    W_px, H_px = project_marker_size(params.W_probe, params.H_probe,
```

```
                                    D_min, params.f, params.p)
A_proj_px = W_px * H_px
A_image_px = 1936 * 1216
A_frac = A_proj_px / A_image_px

IF A_frac > 0.25 THEN
    RAISE ERROR: "Marker projection too large"
END IF

// Initialize candidate list
candidates = []
skipped_printability = 0

// Nested loop over design variables
FOR sigma IN [0.6, 0.7, 0.8, ..., 3.0]:
    FOR r IN [1, 2, 3, ..., 12]:
        // Step 1: Compute minimum pixel size required for detection
        kernel_size = 2 * ceil(3 * sigma) + 1
        patch_size = 2 * r + 1
        s_min_pixels = max(kernel_size, patch_size)

        // Step 2: Map to required physical cell size at D_max
        D_max = 350 mm  // farthest working distance
        feature_size_mm = (s_min_pixels * 0.00345 * 350) / 4

        // Step 3: Check printer constraint
        IF feature_size_mm < 0.06 mm THEN
            skipped_printability = skipped_printability + 1
            CONTINUE  // skip this candidate
        END IF

        // Step 4: Compute grid counts
```

```
G_x = floor(120 / feature_size_mm)
G_y = floor(130 / feature_size_mm)

// Step 5: Evaluate square grid geometry
N = min(G_x, G_y)
N_IDs_square = (N+1)^2
allow_square = FALSE

IF N > 0 AND N_IDs_square >= 40 THEN
    // Check patch proportionality guard
    cell_size_mm_sq = min(130/N, 120/N)
    cell_width_px_min = (4 * cell_size_mm_sq) / (250 * 0.00345)
    required_patch = 0.105 * cell_width_px_min

    IF (2*r + 1) >= required_patch THEN
        allow_square = TRUE
        cell_pitch_square = min(130/N, 120/N)
        APPEND (sigma, r, s_min_pixels, feature_size_mm,
               N, N, N_IDs_square, "square",
               cell_pitch_square) TO candidates
    END IF
END IF

// Step 6: Evaluate rectangular grid geometry
N_IDs_rect = (G_x + 1) * (G_y + 1)
allow_rect = FALSE

IF G_x > 0 AND G_y > 0 AND N_IDs_rect >= 40 THEN
    // Check patch proportionality guard
    cell_width_px_min = (4 * feature_size_mm) / (250 * 0.00345)
    required_patch = 0.105 * cell_width_px_min
```

```
            IF (2*r + 1) >= required_patch THEN
                allow_rect = TRUE
                cell_pitch_rect_x = 130 / G_x
                cell_pitch_rect_y = 120 / G_y
                APPEND (sigma, r, s_min_pixels, feature_size_mm,
                       G_x, G_y, N_IDs_rect, "rect",
                       min(cell_pitch_rect_x, cell_pitch_rect_y)) TO candidates
            END IF
        END IF
    END FOR
END FOR

// Step 7: Compute normalization bounds
N_IDs_min = min(c.N_IDs for c in candidates)
N_IDs_max = max(c.N_IDs for c in candidates)
s_min_px_min = min(c.s_min_pixels for c in candidates)
s_min_px_max = max(c.s_min_pixels for c in candidates)

// Step 8: Score and rank candidates
FOR c IN candidates:
    IF objective == "max_ids" THEN
        c.score = c.N_IDs + (c.s_min_pixels / 1e9)
    ELSE IF objective == "max_robustness" THEN
        c.score = c.s_min_pixels + (c.N_IDs / 1e9)
    ELSE IF objective == "balanced" THEN
        c.N_IDs_norm = (c.N_IDs - N_IDs_min) / (N_IDs_max - N_IDs_min + 1e-12)
        c.s_norm = (c.s_min_pixels - s_min_px_min) / (s_min_px_max - s_min_px_min
        c.N_IDs_norm = clamp(c.N_IDs_norm, 0, 1)
        c.s_norm = clamp(c.s_norm, 0, 1)
        d_utopia = sqrt((1 - c.N_IDs_norm)^2 + (1 - c.s_norm)^2)
        c.score = -d_utopia  // negative distance: closer to utopia = higher score
    END IF
```

```
    END FOR

    SORT candidates BY score (descending)

    // Step 9: Identify top candidate(s)
    best_score = candidates[0].score
    top_candidates = [c for c in candidates if abs(c.score - best_score) < 1e-10]

    RETURN {
        "best": top_candidates[0],
        "all_candidates": candidates,
        "top_tied_count": len(top_candidates),
        "skipped_printability": skipped_printability
    }
END FUNCTION
```

A.5.2 Probe Marker Optimization Results

The optimization was run with the following configuration for the ultrasound probe marker (130 mm × 120 mm, 4 mm focal length, working distance 250–350 mm):

Balanced Compromise Solution The balanced objective prioritizes both identification capacity and detection robustness. The optimal configuration is:

- Gaussian smoothing parameter range: $\sigma \in [5.10, 5.30]$
- Corner detection patch parameter range: $r \in [3, 12]$
- Minimum feature size (pixels): 33
- Required cell size (mm): 9.9619
- Grid configuration: Rectangular, $G_x = 13$ columns, $G_y = 12$ rows
- Actual cell pitch (mm): 10.0000
- Total unique corner IDs: 182 (computed as $(13 + 1) \times (12 + 1) = 14 \times 13 = 182$)

- Marker area fraction: 16.18% of the camera's field of view
- Cell size in pixels at closest distance: 473 pixels per cell (well above noise floor)

This balanced solution provides excellent redundancy for tracking (182 corners) while maintaining sufficient feature size (33 pixels) for reliable detection across the working range. This configuration was selected as the initial marker design for experimental validation, as it represents an optimal balance between identification capacity and robustness that was expected to support reliable pose estimation under clinical operating conditions.

Design Refinement: Transition to Maximum Identification Capacity During experimental validation of the balanced configuration, systematic pose estimation accuracy testing revealed degraded rotational accuracy around the beam axis. In response to these empirical findings, the optimization objective was shifted from *balanced compromise* to *maximum identification capacity*. The rationale is that higher corner density would provide superior geometric conditioning for the pose estimation problem, enabling the RANSAC and refinement stages to more robustly estimate all six pose parameters even under challenging imaging conditions.

Maximum Identification Capacity Solution Prioritizing raw identification capacity yields:

- Gaussian smoothing parameter: $\sigma = 4.00$
- Corner detection patch parameter range: $r \in [3, 12]$
- Minimum feature size (pixels): 25
- Required cell size (mm): 7.5469
- Grid configuration: Rectangular, $G_x = 17$ columns, $G_y = 15$ rows
- Actual cell pitch (mm): 7.6471
- Total unique corner IDs: 288 (computed as $(17 + 1) \times (15 + 1) = 18 \times 16 = 288$)
- Marker area fraction: 16.18% of image
- Cell size in pixels at closest distance: 362 pixels per cell

This configuration was implemented and validated experimentally. The increased corner count (288 total, particularly 15 rows providing much richer y-axis constraint) successfully resolved the accuracy limitations observed with the balanced design.

Maximum Identification Capacity Solution (Final Design) To address the clinical practicality concern while retaining the accuracy benefits of higher corner density, the marker dimensions were reduced. The optimization was re-run with the smaller marker size (reduced from the initial to 80 mm × 50 mm envelope). The final optimal configuration under the maximum identification capacity objective is:

- Gaussian smoothing parameter: $\sigma = 4.00$
- Corner detection patch parameter range: $r \in [4, 12]$
- Minimum feature size (pixels): 25
- Required cell size (mm): 7.5469
- Grid configuration: Rectangular, $G_x = 10$ columns, $G_y = 6$ rows
- Actual cell pitch (mm): 8.0000
- Total unique corner IDs: 77 (computed as $(10 + 1) \times (6 + 1) = 11 \times 7 = 77$)
- Marker area fraction: 4.15% of image (significantly more conservative)
- Cell size in pixels at closest distance: 378 pixels per cell

A.6 Optimization Algorithm: Patient Marker

With the patient marker dimensions obtained from the CAD prototype, the optimization procedure described for the probe marker (Section A.5) was applied identically to the patient marker. The same optical parameters and working envelope were used: $f = 4$ mm, $p = 0.00345$ mm/pixel, and $D_{\min} = 250$ mm to $D_{\max} = 350$ mm.

A.6.1 Patient Marker Optimization Results

The patient marker optimization produced the following best configuration under the `max_ids` objective with rectangular grid geometry:

Maximum Identification Capacity Solution (Patient Marker)

- Objective: `max_ids`
- Grid geometry: Rectangular
- Working distance range: 250–350 mm
- Gaussian smoothing parameter range: $\sigma \in [4.10, 4.30]$
- Corner detection patch parameter range: $r \in [4, 12]$
- Minimum feature size (pixels): 27
- Required cell size (mm): 8.1506
- Grid configuration: $G_x = 7$ columns, $G_y = 9$ rows
- Actual cell pitch (mm): 8.3333
- Total unique corner IDs: 80 (computed as $(7 + 1) \times (9 + 1) = 8 \times 10 = 80$)
- Marker area fraction: 4.67% of the camera's image area
- Cell size in pixels at closest distance: 394 pixels per cell

A.7 Working Distance and Constraint Evaluation Strategy

The optimization must account for the full operating range $[250 \text{ mm}, 350 \text{ mm}]$ at which the marker will be used during clinical procedures. The two key constraints are evaluated at different distances to ensure feasibility across the envelope:

- **Image-area constraint** (Section A.1) is evaluated at the closest distance ($D_{\min} = 250$ mm), representing the worst-case (largest) marker projection. This ensures the sonographer retains adequate field of view even when the marker is closest.
- **Detectability constraint** (Section A.3) is evaluated at the farthest distance ($D_{\max} = 350$ mm), representing the worst-case (smallest) projected feature size. This ensures features remain visible even at maximum distance.

This dual-distance strategy ensures the design is valid across the entire working envelope: the marker does not exceed the image-area budget when closest, and features remain detectable even at maximum distance. Intermediate distances within the 250–350 mm range are guaranteed to satisfy both constraints due to monotonic behavior of projection size with distance.

A.8 Patch Coverage Proportionality

A critical practical consideration is ensuring the detection patch size is proportionate to the projected cell size at the working distances. If the patch is too large relative to the cell, the local neighborhood sampled during corner detection becomes dominated by background, reducing the ability to discriminate true corners from noise or spurious edges.

Consider the detection algorithm's perspective: it searches for corners by analyzing gradients within a $(2r+1)\times(2r+1)$ pixel neighborhood. For this analysis to be meaningful, the patch should be comparable in size to the feature being detected. A patch that is much larger than the projected cell size will average over multiple cells or large background regions, smearing the corner response and degrading localization accuracy.

To enforce proportionality, the optimization requires that the patch size (in pixels) cover a minimum fraction of the projected cell width at the closest working distance:

$$(2r+1) \geq f_{\text{patch}} \times w_{\text{cell,px}}(D_{\min}) \tag{A.22}$$

where:

- $f_{\text{patch}} = 0.105$ is the minimum coverage fraction (empirically determined)
- $w_{\text{cell,px}}(D_{\min})$ is the projected cell width in pixels at the closest distance $D_{\min} = 250$ mm
- $(2r+1)$ is the patch size in pixels

Rearranging in terms of physical cell size:

$$2r+1 \geq 0.105 \times \frac{f \cdot \text{cell\_size\_mm}}{D_{\min} \cdot p} \tag{A.23}$$

The threshold $f_{\text{patch}} = 0.105$ ensures that even large patches ($r = 12$, yielding $2r + 1 = 25$ pixels) remain well-proportioned to realistic cell sizes. In practice, this constraint filters out configurations where the patch is too large for the cell geometry, leaving only those where the detection window is appropriately scaled to the feature size.

A.9 Algorithm Outputs and Candidate Selection

Upon completion of the parameter sweep and candidate ranking, the optimization algorithm produces several key outputs to inform the final marker design decision:

A.9.1 Ranked Candidate List

All feasible configurations are ranked by score according to the selected objective function. This ranked list allows examination of trade-offs:

- The top candidate represents the algorithm's optimal recommendation under the chosen objective
- Alternative high-scoring designs may be explored if empirical validation suggests modifications are needed
- The ranked list reveals the Pareto frontier of efficiency

A.9.2 Best Configuration Summary

The algorithm identifies and highlights the single best scoring candidate(s) under the chosen objective. This recommendation specifies:

- The (σ, r) parameter pair(s) achieving the optimal score (multiple pairs may tie due to discrete quantization)
- The grid configuration (number of cells in each direction: G_x, G_y)
- The total number of unique corner IDs supported
- The minimum pixel size required for robust detection
- The corresponding physical cell size that must be printed
- The marker area fraction as a percentage of the camera's field of view

A.9.3 Constraint Violation Analysis

The algorithm verifies that all selected candidates satisfy the feasibility constraints. Diagnostic counts are reported for each constraint type:

- Number of configurations filtered by printer resolution limit
- Number of configurations violating minimum ID requirement
- Number of configurations failing patch coverage proportionality guard

This diagnostic information helps understand which constraints are binding and which are relaxed, guiding future sensitivity analysis if design objectives change.

A.10 Integration with Marker Fabrication and Detection

The outputs from the optimization algorithm serve as the starting point for implementation:

1. Marker Fabrication: Print the marker using the specified cell size (mm) and grid geometry. The final design specifies a regular rectangular grid with cell pitch of 8.0 mm (for the final probe marker) or 10.0 mm (for the balanced solution), ensuring all cells are identical and printable at the target DPI.

2. Corner Detection Algorithm Tuning: Set σ and r parameters in the corner detection algorithm according to the recommended values (e.g., $\sigma = 4.0$ and $r \in [4, 12]$ for the final probe design). These parameters control Gaussian smoothing aggressiveness and patch size, directly affecting detection sensitivity and localization accuracy.

3. Empirical Validation: Test corner detection on the printed marker across the full working distance range (250–350 mm) in realistic lighting conditions to confirm robustness matches theoretical predictions. Compare detected corner count and localization repeatability against the optimized expectations.

4. Refinement: If validation reveals poor detection performance (e.g., significant corner loss at maximum distance, or high jitter), examine alternative high-scoring candidates from the ranked list, or adjust constraints and re-run the optimization.

The design optimization thus provides a principled starting point grounded in optical physics, detectability principles, and manufacturing constraints. However, practical validation in realistic imaging conditions remains essential to confirm algorithm performance and margin against real-world variability in lighting, marker degradation, and camera calibration drift.

A.11 Software Implementation and Reproducibility

The complete optimization algorithm is implemented in the Python module `OptimizeMarkerSize.py` and is publicly available on GitHub at https://github.com/samuv01.

Appendix B

Guide on how to use the HydraMarker Pipeline

This document describes the complete workflow to:

1. Design, generate and print a custom marker.
2. Calibrate the camera for the specific working distance.
3. Record mapping and motion data.
4. Generate a virtual map of the marker.
5. Run the detection, identification and tracking of the marker.
6. (Optional) Compute transforms between the marker map and the real tool.
7. (Optional) Print HydraMarker on paper or vinyl sticker.

The goal is that anyone can reproduce the full pipeline **without getting lost / asking questions**. In case you have questions you can send me an email at vaccari.samu@gmail.com (version of 01/2026).

Feel free to make changes to the pipeline but make sure to add them to this guide with a different color and include your contact below:

-Santa Claus 12/2026 (santaclaus@ubc.com);

1. Design the Marker

Script: OptimizeMarkerSize.py

Goal: choose the best marker layout for your experiment.

From now on we're going to talk about Marker and IDs. An ID is a corner of the checkerboard

1. Set the main parameters:

- Marker dimensions (width and height);
- Pixel size of the sensor you're working with;
- Working distance (WD) (set value or range, usually better to set range);
- Focal length of the hardware you're using;
- Image resolution (width and height in pixels);
- Maximum image fraction of the marker in the camera field of view at the set WD;
- Minimum number of IDs required (ID = Corner of the checkerboard);
- Search ranges for **sigma** and **r** (detection parameters);
- Minimum printable feature size and printer DPI;
- Objective of the optimization:
  - Max IDs (smaller checkers so more corners used for the pose estimate)
  - Max Robustness (bigger checkers, easier to detect but less corners for the pose estimate)
  - Balanced (balanced solution between the previous 2)
  - Printer Info (default ones are CHHM 7^{th} Floor (XEROX ALTALINK C8055))

2. Output: MAKE SURE TO WRITE THESE DOWN BEFORE PROCEEDING

- **sigma** value or range (detector parameter);
- **r** value or range;
- actual_cell_mm (physical spacing between cells);
- grid_x × grid_y (marker grid dimensions);

- cell_px (cell size in pixels for printing (already accounting for printer DPI)).

2. Generate the Marker and pattern file

Script: GenerateMarker_SV.py.

1. The marker field and tag shapes are defined:

- field is a 2D array setting the main grid size (e.g. 9×9). You have to use what you obtained from the OptimizeMarkerSize.py as marker grid dimensions;
- tag_shapes is a list of binary masks describing different tag shapes (e.g. 3×3 and 1×9).

   (If you don't know what these parameters are please read Zhu et al. (2023))

2. Set the cell-size obtained from the optimizer.
3. Modify the padding if you want, but leave some so that the detector can identify the outer corners of the grid.
4. Output: a subfolder is created named from the field and tag shapes with the following:
    - marker_board.pdf (marker on a letter size for printing)
    - marker_sta.npy (binary marker pattern used throughout the pipeline);
    - marker_board.png (marker image);
    - marker_field (generated field)

If you want to be ecofriendly (and you should) use duplicate_marker_in_PDF.py to upload the png and create a letter size pdf with multiple markers.

5. Print the marker using a high quality printer (CHHM 7th floor).

Make sure to print at **1200 dpi** and **NO SCALING**. **After printing measure the dimension of the grid (excluding the padding) and of a cell.** These should match respectively what you set in the optimizer and what you obtained as actual_cell_mm.

If they don't match repeat from the beginning.

6. Attach the marker to the tool you want to track. It's paper so bend it as much as possible, you can also cut pieces of the marker off to better adhere to the surface.

3. Calibrate the camera

For this step I did not write a code because you should have it for your particular camera. After mounting the camera on a stable tripod acquire calibration images covering different orientations and positions. Make sure to maintain the calibration chessboard at the working distance you're planning to use the marker. Check that the reprojection error is low (less than 1 px – e.g I was happy with 0.2 px). Save the intrinsics as camera_mtx.npy and camera_dist.npy in a fixed folder.

4. Record Mapping and Motion Data

The pipeline expects **.npz recordings** containing image data and timestamps. While the acquisition code is not part of this manual, the following conventions are recommended:

4.1 Mapping Recording

Goal: build a robust point cloud of the IDs of the marker wrapped around the tool (3D map of the marker). To accomplish this, structure from motion is used.

- The tool is **on a rotating stage**;
- The camera is fixed on a stable tripod at the WD;
- Start recording with the tool perpendicular to the line of sight of the camera and hold it still for a couple of seconds (this is good for the bootstrap step) then SLOWLY rotate the tool; Make sure to rotate it for the full width of the marker.
- Save the recording as a single .npz file (e.g. mapping_recording.npz).

This file will be referenced as mapping_npz in AutomationConfig if a map does not yet exist.

4.2 Motion Recording(s)

- Typically, the camera is fixed, while the tool moves according to the protocol;
- For each experimental condition, record a separate .npz file;
- Store these motion recordings in a dedicated folder.

This folder will be used as motion_npz_dir in AutomationConfig.

The mapping step is fundamental and it can be considered as the calibration/registration of the tool. Before leaving the lab and packing everything up because you recorded your trajectories, you should make sure that the map is built correctly (otherwise you have to redo it). To do this you can run proceed to step 5 and interrupt the code once the map is created or do the same manually by running the detection (read_marker_fromNPZ.py) and the structure from motion (sfm_runner.py).

5. Detect, Identify and Track the tool

Script: Orchestrator.py

The AutomationConfig dataclass defines all inputs required by the automation.

For typical use, the following fields must be edited in the __main__ block:

- sta_and_map_dir (folder created by GenerateMarker_SV.py containing marker_sta.npy);
- motion_npz_dir (folder containing the motion .npz recordings);
- mapping_npz (.npz where you rotated the tool (set to None if the map already exists in the sfm_outputs folder));
- camera_matrix_path, distortion_path (paths to camera_mtx.npy and camera_dist.npy);
- pipeline_script (path to ReadMarker_SV.py);
- cell_size_mm (set to actual_cell_mm from the optimizatio);
- r, sigma (set to the selected values from the optimization).

In the AutomationConfig dataclass there is also:

```
# Marker IDs used to define object frame in Structure from Motion

origin_marker_id: int = 17

x_axis_marker_id: int = 59

y_axis_marker_id: int = 21
```

This is where you define where you want your origin to be and how you want the axis of the frame of the marker to be oriented. The Z axis comes from the right hand rule.

Numbering of the IDs in the marker.

You're probably going to ask how you could know the numbering of the ids or obtain the figure shown above, before even mapping and tracking. There are 2 options:

1. You count all the corners from your png (excluding the padding) starting from 1 and going down through the rows;

2. (Preferred) You take single pictures and quickly run a detection and identification with ReadMarker_FromNPZ.py. Make sure to set the directories properly in the __main__ block and the right r and sigma in the config class at the top. To obtain the overlay like the one above turn save_overlays: bool = True. Make sure to turn it back to false after running so that you don't save all the overlays for all the recordings that you're going to process later.

In the script you should also modify this line with a proper number of minimum shared markers (I meant to write IDs).

```
filter_cfg = FilterConfig(min_shared_markers=40,min_bootstrap_inlier_ratio=0.8
```

This value is used for the generation of the map and in particular the bootstrap step. This step consists of choosing 2 frames that have at least the minimum amount of shared

IDs, and building the map by triangulation with these 2 frames as "starting point". The result of this initial rough triangulation is used to give a good starting point to the Bundle Adjustment. With this optimization we try to minimize the total reprojection error. If the recording was done properly the optimization should end with a very low value (in my case: Median reprojection error (post-BA): 0.064 pix, Mean reprojection error (post-BA): 0.079 pix).

NOTE: my thresholds for the first triangulation are various and very tight. I'd rather discard a lot of frames than having outliers when trying to calibrate my marker map on the tool.

The Orchestrator.py has a class to track multiple tools at the same time named MultiMarkerAutomationConfig. In order to run this you need to generate 2 or more markers. To make sure that the markers are unique I suggest perturbing some cells in GenerateMarker_SV.py. This can be done by uncommenting:

```
perturb_unknown_cells(field, fraction=PERTURB_FRACTION,seed=PERTURB_SEED)
```

and setting its variables. This is necessary since sometimes the generator would come up with the same solution for different runs.

Once these are generated, printed and attached to the tool, you have to map each tool separately first, as described in the section 4.1. With the recordings you have to run the detection and identification with ReadMarker_FromNPZ.py. At the bottom of this code **set the starting ID (id_base) to 1001 for 1st tool, 2001 for 2nd tool and so on**. Then use the sfm_runner.py to create the virtual map. Make sure to properly set the origin_marker_id, x_axis_marker_id, y_axis_marker_id, cell_size_mm following the id_base previously set. Move the outputs in a subfolder per each marker folder called mapping and import the sfm outputs. You should have a main folder with a subfolder for each marker with its information (sta matrix and mapping). Record the motion with both tools in the camera frame and indicate the correct directory in the config settings of the Orchestrator.py.

6. Calculate the transform from marker origin to tool origin

In the Orchestrator.py you could also upload a transform from the origin of the marker map to the origin of the frame of the tool that you're tracking.

Turn on the NDI Vega so it warms up.

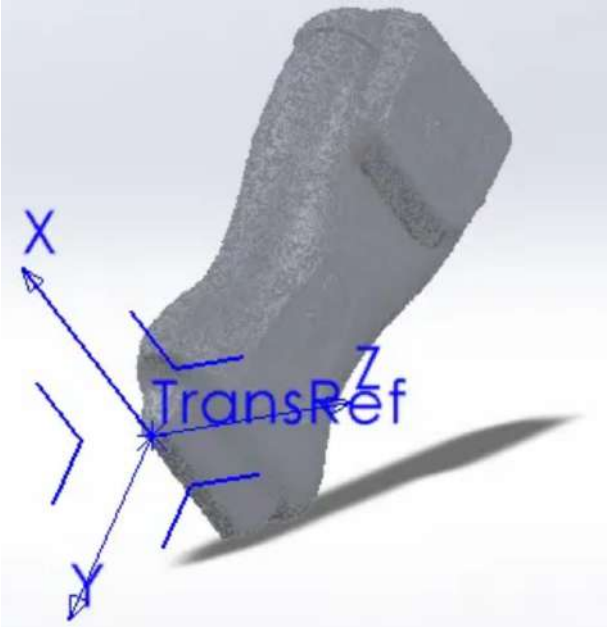
Before doing step 6. make sure your marker map (MM) is the same as the mapping step. If there are tears/the physical marker moved/the camera was used by someone else and they changed the f# or the focus, repeat the calibration step and the mapping step.

Once you have a proper MM, use the script map_for_slicer.py. This code takes the .npz file generated by the SfM, so the 3d map of the object and converts it into a .mrk.json and a .fcsv. Use from CLI specifying the correct PATHS:

python map_for_slicer.py marker_map_aligned.npz

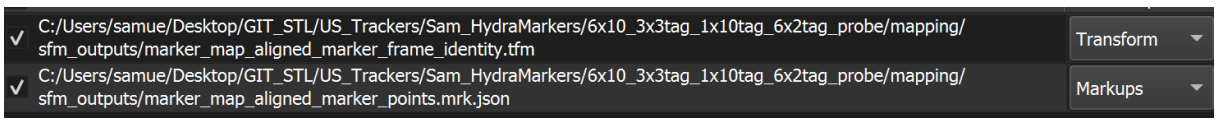
The files are generated in the same folder as the marker_map_aligned.npz.

You need a 3D model of the tool, in particular an .STL file that has the origin and the reference frame oriented as you need. If you're not sure what your reference frame is, use a CAD software to locate it and eventually move it. When you export it, make sure to set as reference frame of the .STL, the reference frame you just generated. I moved it to the origin of the transducer and oriented as needed as you can see in the picture below.



STL file with the correct reference frame.

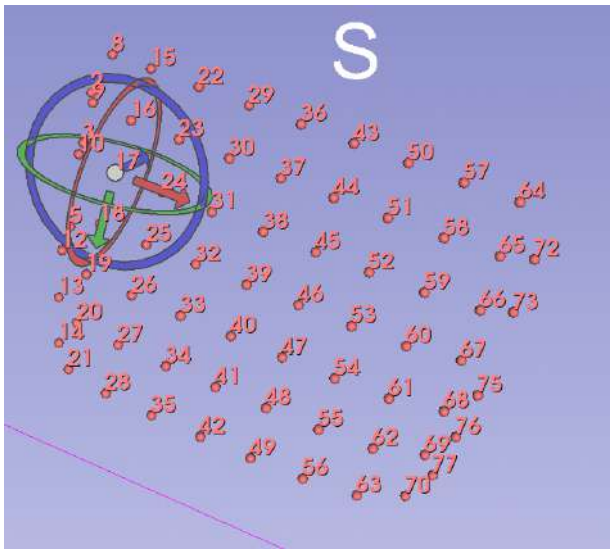
Start by loading the MM on slicer and the identity transform.



Loading the MM and the identity transform on Slicer.

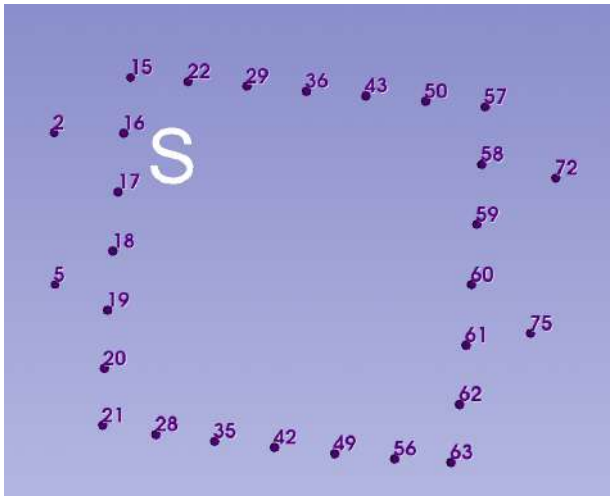
Rename the Transform as MarkerToTracker and drag the MM on top of it so that it

becomes its child.



MM on Slicer.

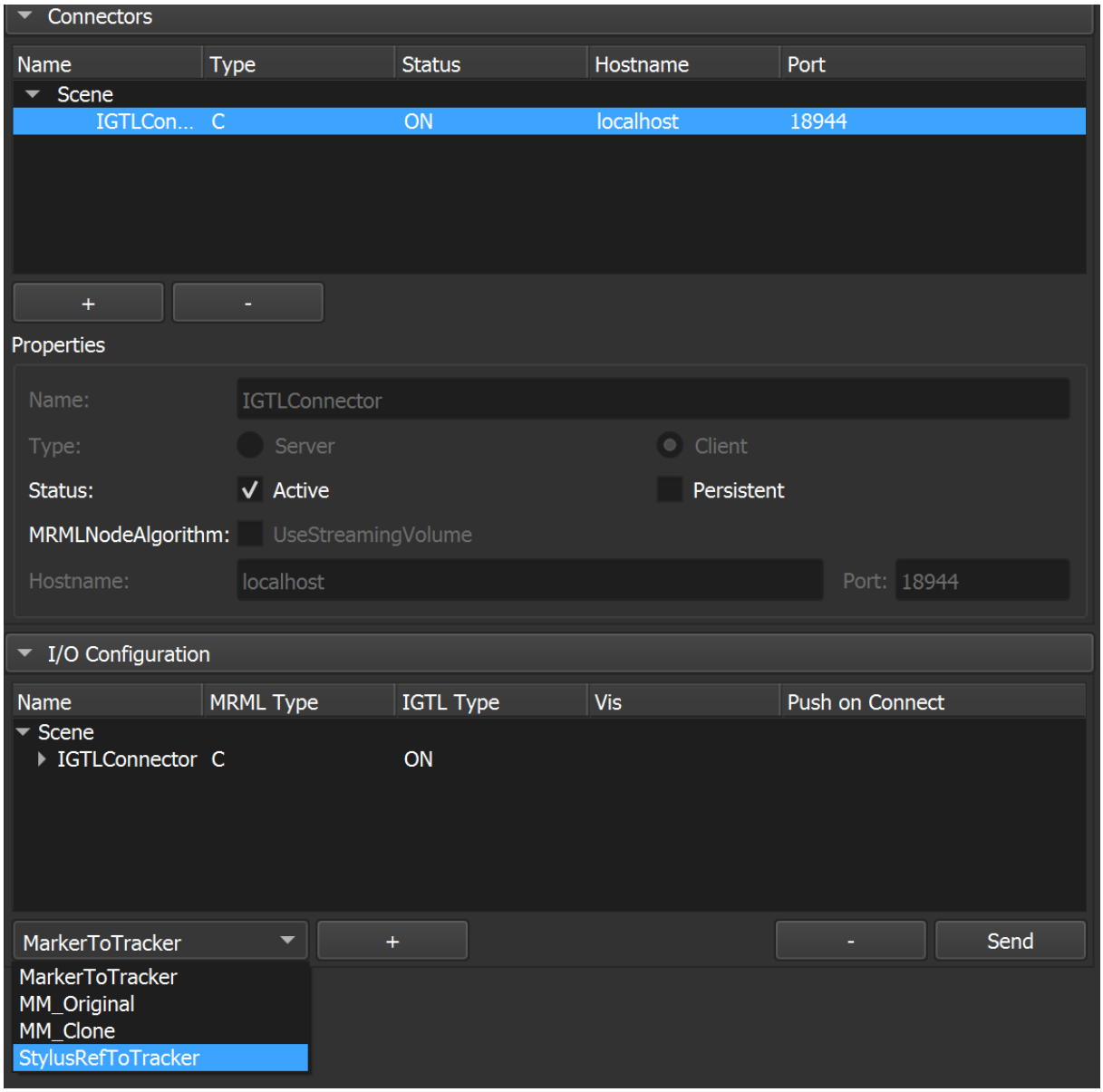
Generate a copy of the MM, drag it on MarkerToTracker and lock the control points of the original one. Take the copy and start deleting ids, if you have a lot like I do. Slicer can work with maximum 30 control points (IDs) in the next steps. Hide the original MM for clarity (also change color so you know what you're working with). The subset you keep should have IDs on all sides of the MM (see id 2, 5, 72, 75).



Subset of the MM.

Lock the final control points so you can't move or delete the points, rename as MM_Clone.

Use IGT Connect and the Plus Server to use the NDI on Slicer. Make sure to have the stylus in your config file.



IGT Connect interface on Slicer.

Make sure the Transform StylusRefToTracker is visible in the list at the bottom. It has to be in the field of view of the NDI in order to be displayed.

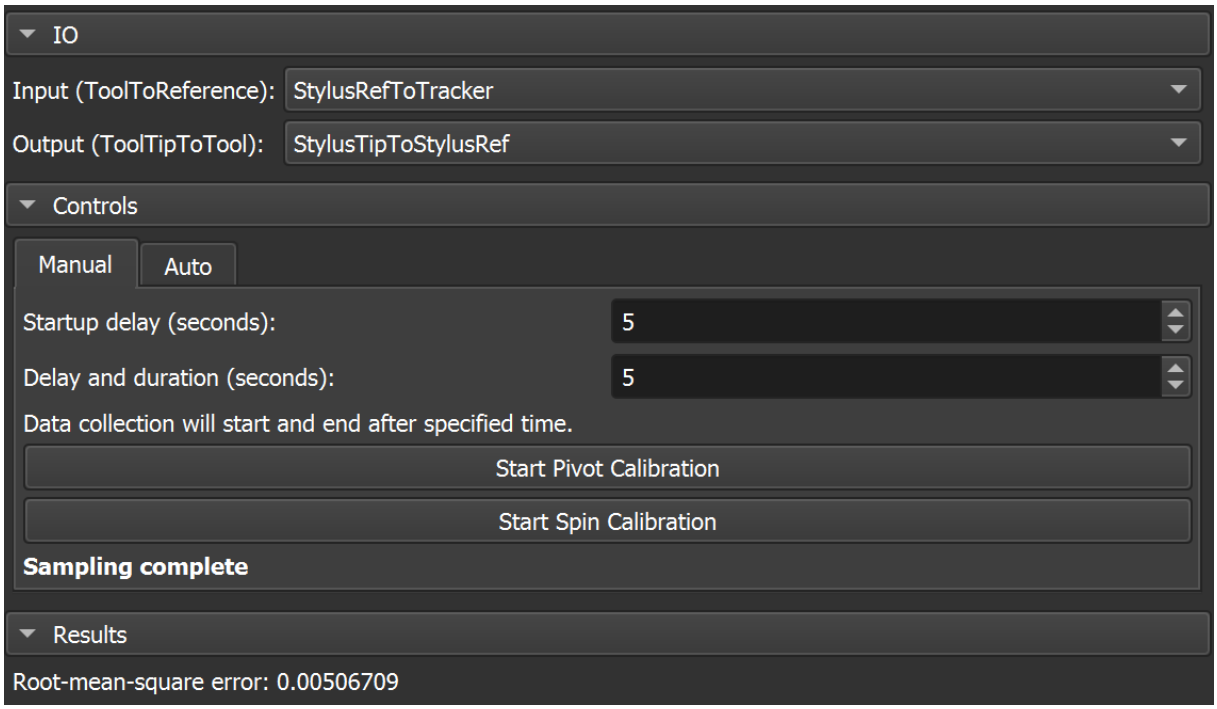
Perform Stylus Calibration *(From "Mandibular Reconstruction: Software Operator Instructions")*:

1. Using the module menu, navigate to the IGT modules and select "Pivot Calibration."
2. Select *Input transform* as `StylusRefToTracker`.
3. Select *Output transform* as `StylusTipToStylusRef`.
4. Choose startup delay and duration (5 s each is default and acceptable).

5. Select “Start Pivot Calibration” and rotate the stylus around its tip, keeping the tracker facing the NDI Vega camera.

6. Select “Start Spin Calibration” and spin the stylus about its long axis (the tracker on the probe will not always face the camera).

7. The resulting calibration error should be <0.1 mm.

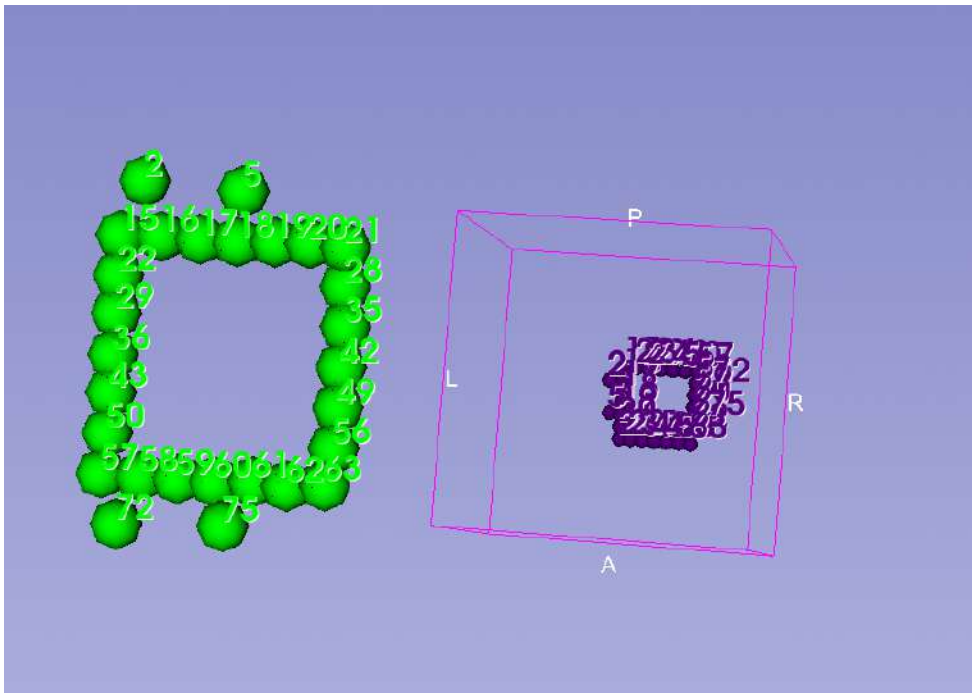


Stylus Pivot Calibration interface on Slicer.

Put the stylus tip to stylus ref as child of stylusreftotracker.

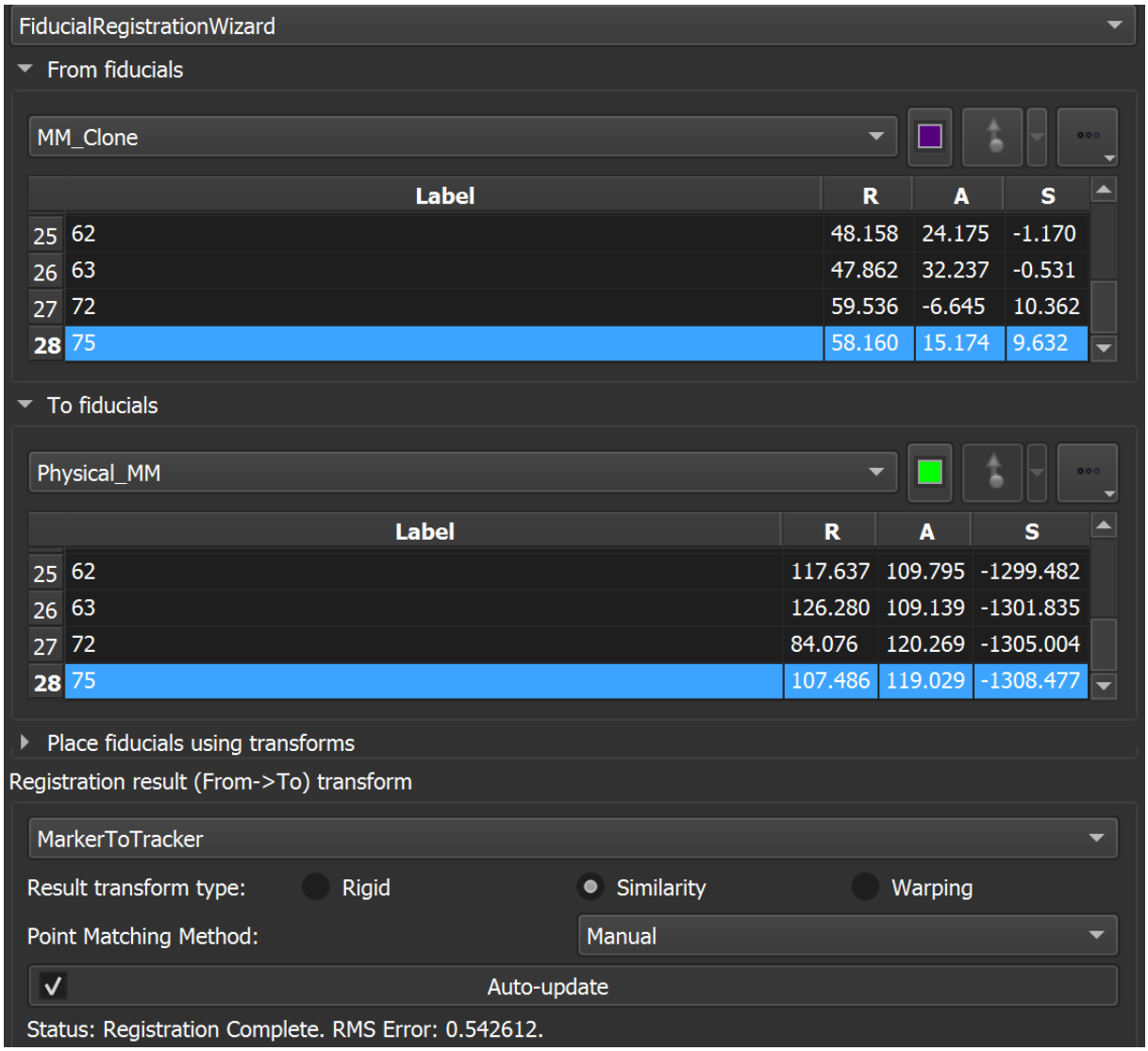
Place your tool in a FIXED/CLAMPED setup. It’s important that it is fixed throughout the whole procedure.

Using the module menu, navigate to the IGT modules and select “Collect Points”. Create a new markup list of points and name it Physical_MM. Set as sampling node the StylusTipToStylusRef. Put the tip on each of the ids of the MM_Clone and collect each point.



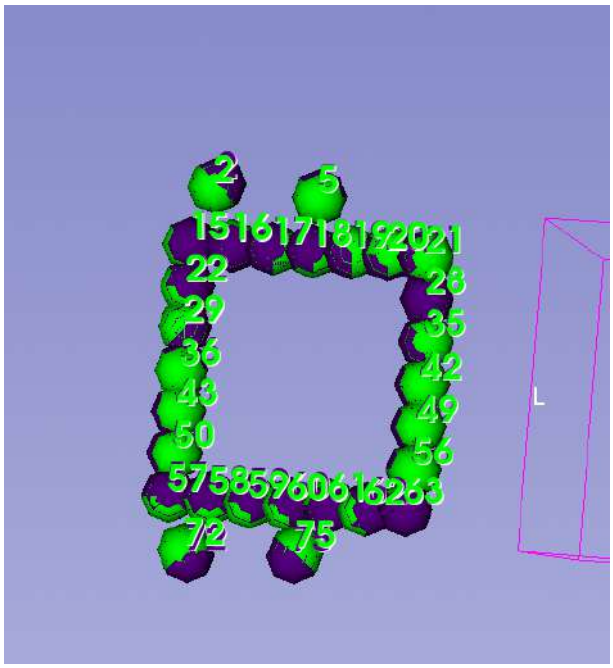
Collected points in green and MM_Clone in purple.

Using the module menu, navigate to the IGT modules and select “Fiducial Registration Wizard”. Set it as shown below. Overwrite the results to the MarkerToTracker Transform.



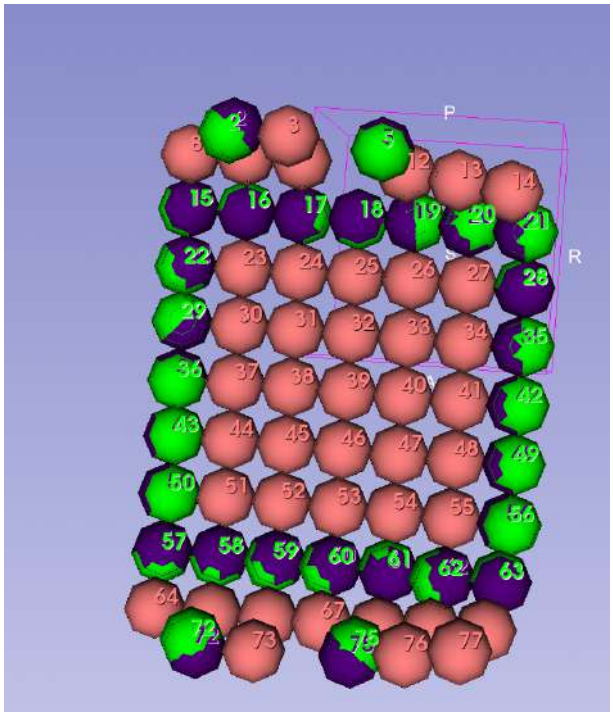
Fiducial Registration Wizard interface on Slicer while registering the MM_Clone.

And if you apply this transform you will see the MM_Clone moving on top of the points that we just digitalized.



MM_Clone (purple) on top of the physical points (green).

And if you make the whole original MM visible:



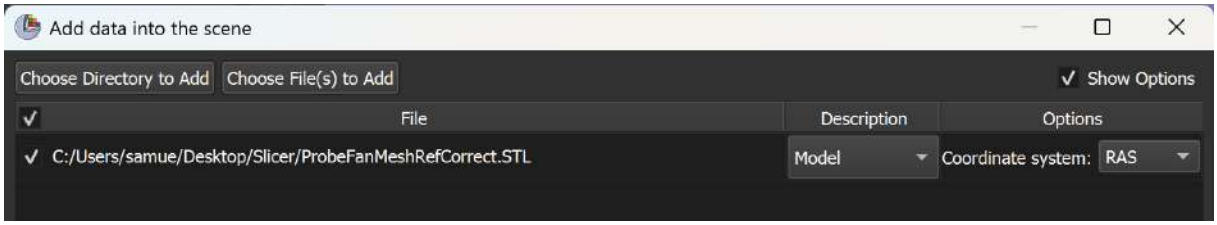
MM_Clone (purple) and original MM (pink) on top of the physical points (green).

If they don't overlay two things might have happened:

1. You did not collect the points properly -> recollect the points

2. You did not set the MM as children of the MarkerToTracker

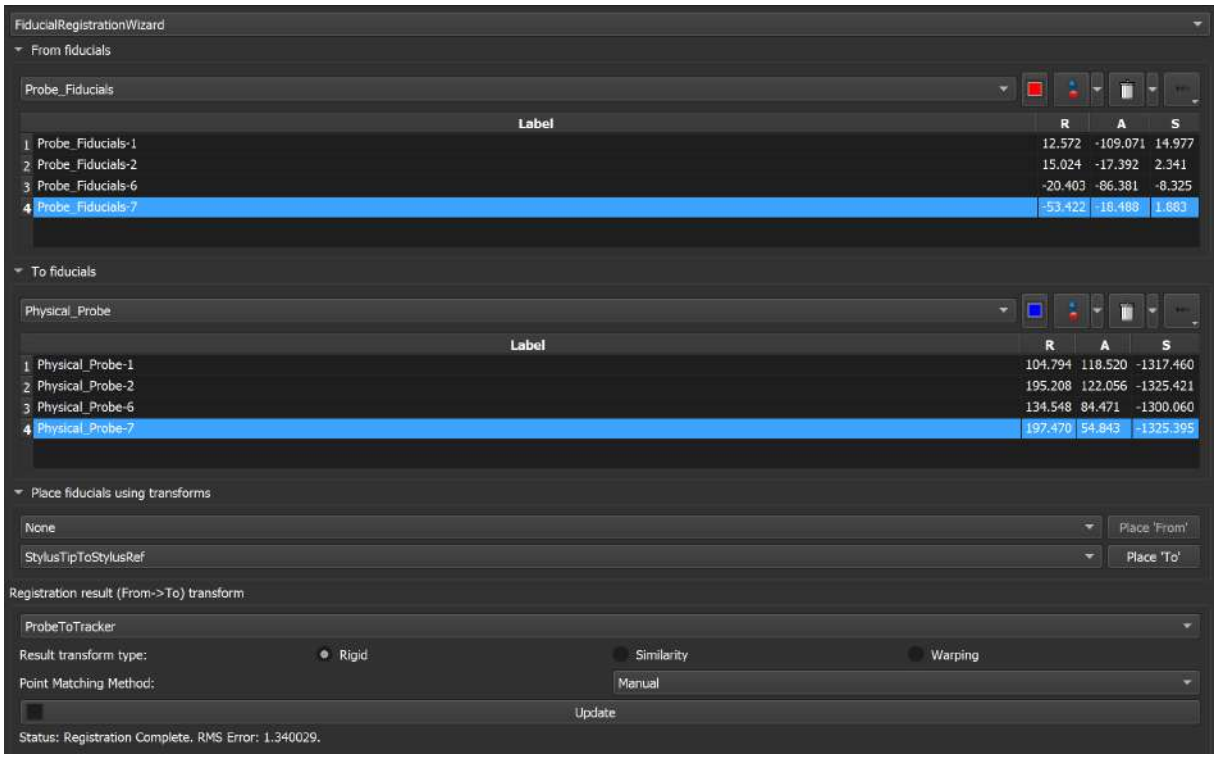
Now upload the .STL with the characteristics I previously described and set the coordinate system to RAS.



STL file upload.

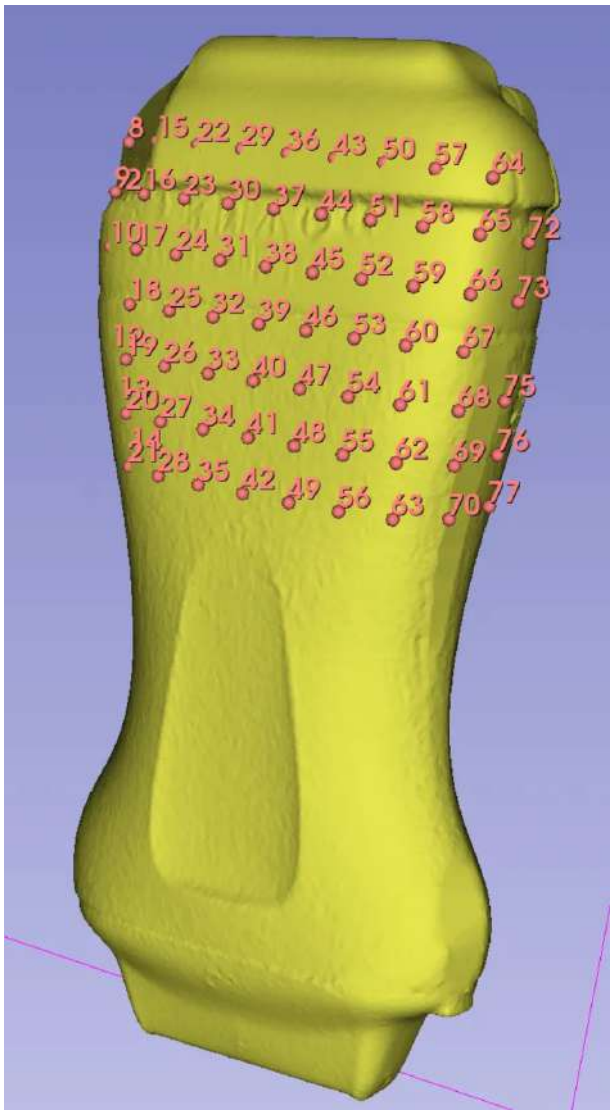
Using the module menu, navigate to the IGT modules and select “Fiducial Registration Wizard”.

Add manually points to the .STL model, under a new markup (probe_fiducials). These should be non collinear and in locations that are easy to identify both on the model and the tool. Place fiducials using the stylus tip as sampling node. Create a new transform and name it ProbeToTracker. I looked for an error below 1.5mm.



Fiducial Registration Wizard interface for registering the STL.

If you want you can put the .STL as child of ProbeToTracker by dragging and dropping it, to see if the MM is actually on the model.



MM on the STL.

The last step is to open the Python Interactor on slicer and run this code (make sure to edit the PATH at the bottom):

```
import slicer, vtk, numpy as np

# Get transform nodes (use the node names as strings)
markerToTracker = slicer.util.getNode("MarkerToTracker")
probeToTracker  = slicer.util.getNode("ProbeToTracker")

# Get their 4x4 matrices
m_T_T_M = vtk.vtkMatrix4x4()
markerToTracker.GetMatrixTransformToParent(m_T_T_M)
```

```
m_T_T_P = vtk.vtkMatrix4x4()
probeToTracker.GetMatrixTransformToParent(m_T_T_P)

# Invert:  M_T_T = inv(T_T_M)
m_M_T_T = vtk.vtkMatrix4x4()
vtk.vtkMatrix4x4.Invert(m_T_T_M, m_M_T_T)

# Compose: M_T_P = M_T_T * T_T_P
m_M_T_P = vtk.vtkMatrix4x4()
vtk.vtkMatrix4x4.Multiply4x4(m_M_T_T, m_T_T_P, m_M_T_P)

# Convert to NumPy
T_M_P = np.zeros((4, 4), dtype=float)
for i in range(4):
    for j in range(4):
        T_M_P[i, j] = m_M_T_P.GetElement(i, j)

print("T_M_P =\n", T_M_P)

# Save to .npy file (EDIT THIS PATH!)
out_path = r"C:\Users\samue\Desktop\Slicer\FinalProbeResults\T_M_P_from_slicer.npy"
np.save(out_path, T_M_P)

print("Saved M_T_P to", out_path)
```

6. Print HydraMarker on paper or vinyl sticker

I named the chapter *"Optional"* but maybe not. I think you'll have to print the marker at least once, especially because you're so excited to try it after you designed it!

The quickest and easiest way to try the marker is by printing the PDF that was generated by GenerateMarker_SV.py.

In case you want to create a vinyl sticker so that you can fix it on your tool, you have to use the script duplicate_marker_in_PDF.py, in particular the last block. This allows you

to gather and arrange multiple markers in a letter size pdf. In addition to that there is a second pdf containing the die lines for the cut around each marker sticker. You can send these two to a printing place (e.g. PrintPrint on Pender St) and get as many stickers as you can fit in a letter size page for only 40 CAD. Make sure to let the store know that you don't want the picture scaled, printed at 1200dpi and, with the extra lamination for durability.

Bibliography

Adámek, R., P. Vávra, and A. Babinec (2023). “Analytical Models for Pose Estimate Variance of Planar Fiducial Markers for Mobile Robot Localisation”. In: *Sensors* 23.12, p. 5746. DOI: 10.3390/s23125746.

Adams, S. J. et al. (2021). “Access to ultrasound imaging: A qualitative study in two northern, remote, Indigenous communities in Canada”. In: *International Journal of Circumpolar Health* 80.1, p. 1961392. DOI: 10.1080/22423982.2021.1961392.

Al Azri, M. S., M. N. Al Abdali, and T. A. Taqi (2023). “Epidemiology of developmental dysplasia of the hip at a tertiary hospital in Oman”. In: *Journal of Musculoskeletal Surgery and Research*. DOI: 10.25259/JMSR_45_2023.

Atalar, H. et al. (2007). “Indicators of successful use of the Pavlik harness in infants with developmental dysplasia of the hip”. In: *International Orthopaedics* 31.2, pp. 145–150. DOI: 10.1007/s00264-006-0097-8.

Bakarman, K. et al. (2023). “Developmental Dysplasia of the Hip (DDH): Etiology, Diagnosis, and Management”. In: *Cureus* 15.8, e43207. DOI: 10.7759/cureus.43207.

Barney, J., N. S. Piuzzi, and H. Akhondi (2025). “Femoral head avascular necrosis”. In: *StatPearls*. Available from: https://www.ncbi.nlm.nih.gov/books/NBK546658/. Treasure Island, FL: StatPearls Publishing.

Basler AG (2024). *daA1920-160uc Technical Documentation*. Tech. rep. URL: https://docs.baslerweb.com/daa1920-160uc.

Bencina, R. and M. Kaltenbrunner (2005a). “The design and evolution of fiducials for the reacTIVision system”. In: *Proceedings of the 3rd International Workshop on Gesture and Sign Language Based Human-Computer Interaction*. URL: https://api.semanticscholar.org/CorpusID:13754140.

Bencina, R., M. Kaltenbrunner, and S. Jordà (2005b). “Improved topological fiducial tracking in the reacTIVision system”. In: *2005 IEEE Computer Society Conference on*

Computer Vision and Pattern Recognition (CVPR'05) - Workshops. IEEE, p. 99. DOI: 10.1109/CVPR.2005.475.

Bergamasco, F. et al. (2011). "RUNE-Tag: A high accuracy fiducial marker with strong occlusion resilience". In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 113–120. DOI: 10.1109/CVPR.2011.5995544.

Brown, D. C. (1966). "Decentering distortion of lenses". In: *Photogrammetric Engineering* 32.3, pp. 444–462. URL: https://api.semanticscholar.org/CorpusID:117271607.

Bucher, F. et al. (2021). "Paediatric hip ultrasound: uncertainties in examination and choice of treatment". In: *J Child Orthop* 15.1, pp. 42–47. DOI: 10.1302/1863-2548.15.200084.

Calvet, L. et al. (2016). "Detection and Accurate Localization of Circular Fiducials under Highly Challenging Conditions". In: *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 562–570. DOI: 10.1109/CVPR.2016.67.

Chen, Kexin K. (2025). "Clinical evaluation of optically tracked ultrasound system to support diagnosis of developmental dysplasia of the hip". Retrieved from UBC Open Collections. Master's thesis. The University of British Columbia. URL: https://open.library.ubc.ca/collections/ubctheses/24/items/1.0451193.

Chen, M., R. Cai, et al. (2024). "The diagnostic value of artificial intelligence-assisted imaging for developmental dysplasia of the hip: a systematic review and meta-analysis". In: *Journal of Orthopaedic Surgery and Research* 19. DOI: 10.1186/s13018-024-05003-4.

Clarius Medical (2024). *L7 HD3 Linear Ultrasound Scanner Technical Specifications*. Tech. rep. URL: https://clarius.com/es/scanners/l7/.

Clarke-Grant, D. (June 2025). "Healthcare Access for Indigenous Communities in Rural Canada: A Narrative Review". In: *International Research and Policy Journal* 2025. URL: https://irpj.euclid.int/articles/healthcare-access-for-indigenous-communities-in-rural-canada-a-narrative-review-and-interdisciplinary-framework-for-action/.

Dr Sono (2025). *How much does an ultrasound machine cost*. URL: https://drsono.com/blogs/news/ultrasound-machine-cost/.

Edmund Optics Inc. (2024). *TECHSPEC UC Series Fixed Focal Length Lens #33-300 (4mm FL) Specifications*. Tech. rep. URL: https://www.1stvision.com/lens/edmund-optics/dataman/spec_33300.pdf.

Fan, Z. et al. (2018). "Spatial Position Measurement System for Surgical Navigation Using 3-D Image Marker-Based Tracking Tools With Compact Volume". In: *IEEE Trans Biomed Eng.* 65.2, pp. 378–389. DOI: 10.1109/TBME.2017.2771356.

Fattori, G. et al. (2021). "Technical assessment of the NDI Polaris Vega optical tracking system". In: *Radiation Oncology* 16.1, p. 87. DOI: 10.1186/s13014-021-01804-7.

Fiala, M. (2005). "ARTag, a fiducial marker system using digital techniques". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. Vol. 2. IEEE, pp. 590–596. DOI: 10.1109/CVPR.2005.74.

— (2010). "Designing highly reliable fiducial markers". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.7, pp. 1317–1324. DOI: 10.1109/TPAMI.2009.146.

Fischler, M. A. and R. C. Bolles (1981). "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography". In: *Communications of the ACM* 24.6, pp. 381–395. DOI: 10.1145/358669.358692.

Fitzgibbon, A. W., M. Pilu, and R. B. Fisher (2000). "Direct least square fitting of ellipses". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21.5, pp. 476–480. DOI: 10.1109/34.765658.

Gao, Z., M. Zhu, and J. Yu (2021). "A novel camera calibration pattern robust to incomplete pattern projection". In: vol. 21. 8, pp. 10051–10060.

Garrido-Jurado, S. et al. (2014). "Automatic generation and detection of highly reliable fiducial markers under occlusion". In: *Pattern Recognition* 47.6, pp. 2280–2292. DOI: 10.1016/j.patcog.2014.01.005.

— (2015). "Generation of fiducial marker dictionaries using mixed integer linear programming". In: *Pattern Recognition* 51, pp. 481–491. DOI: 10.1016/j.patcog.2015.09.023.

Geng, C. et al. (2020). "Using 3-dimensional ultrasound islice technology for the diagnosis of developmental dysplasia of the hip". In: *Journal of Ultrasound in Medicine* 39.6, pp. 1117–1123. DOI: 10.1002/jum.15193.

Godward, S. and C. Dezateux (1998). “Surgery for congenital dislocation of the hip in the UK as a measure of outcome of screening. MRC Working Party on Congenital Dislocation of the Hip. Medical Research Council”. In: *Lancet* 351.9110, pp. 1149–1152. DOI: 10.1016/S0140-6736(97)10466-4.

Gorry, P. A. (1991). “General least-squares smoothing and differentiation of nonuniformly spaced data by the convolution method”. In: *Analytical Chemistry* 63.5, pp. 534–536. DOI: 10.1021/ac00005a031.

Graf, R. (2006). *Hip sonography: Diagnosis and management of infant hip dysplasia*. Springer.

Gray, C. J. and M. M. Shanahan (2022). “Breech Presentation”. In: *StatPearls*. URL: https://www.ncbi.nlm.nih.gov/books/NBK448063/.

Gunay, C. et al. (2009). “Correlation of femoral head coverage and Graf alpha angle in infants being screened for developmental dysplasia of the hip”. In: *Int Orthop* 33.3, pp. 761–764. DOI: 10.1007/s00264-008-0570-7.

Gunay, C. et al. (2019). “Can a Hip Diagnosed as Graf Type 1 According to Graf Method Be Normal? A Case Series and Evaluation of the Graf Method”. In: *Journal of Orthopaedic Case Reports* 9.5, pp. 23–27. DOI: 10.13107/jocr.2250-0685.1344.

Hamming, R. W. (1950). “Error detecting and error correcting codes”. In: *The Bell System Technical Journal* 29.2, pp. 147–160. DOI: 10.1002/j.1538-7305.1950.tb00463.x.

Hareendranathan, A. R. et al. (2021). “Artificial Intelligence to Automatically Assess Scan Quality in Hip Ultrasound”. In: *Ultrasound in Medicine & Biology* 47.11, pp. 3460–3469. DOI: 10.1016/j.ultrasmedbio.2021.07.016.

Hareendranathan, A. R. et al. (2022). “Impact of scan quality on AI assessment of hip dysplasia ultrasound”. In: *Journal of Ultrasound* 25.1, pp. 145–153. DOI: 10.1007/s40477-021-00560-4.

El-Hariri, H. et al. (2021). “Automatically Delineating Key Anatomy in 3-D Ultrasound Volumes for Hip Dysplasia Screening”. In: *Ultrasound in Medicine and Biology* 47.9, pp. 2713–2722. ISSN: 0301-5629. DOI: 10.1016/j.ultrasmedbio.2021.05.011. URL: https://www.sciencedirect.com/science/article/pii/S0301562921002416.

Harper, P. et al. (2021). “Cost Analysis of Screening Programmes for Developmental Dysplasia of the Hip: A Systematic Review”. In: *Indian J Orthop* 55.6, pp. 1402–1409. DOI: 10.1007/s43465-021-00501-7.

Hartley, R. and A. Zisserman (2003). *Multiple View Geometry in Computer Vision*. 2nd. Cambridge University Press. DOI: 10.1017/CBO9780511811685.

Hata, K. and S. Savarese (n.d.). *CS231A Course Notes 3: Epipolar Geometry*. URL: https://web.stanford.edu/class/cs231a/course_notes/03-epipolar-geometry.pdf.

Hutchinson, S., G. D. Hager, and P. I. Corke (1996). "A tutorial on visual servo control". In: *IEEE Transactions on Robotics and Automation* 12.5, pp. 651–670. DOI: 10.1109/70.538972.

Island Social Trends (2025). *BC birth tally back up to pre-pandemic level in 2024 & 2025*. URL: https://islandsocialtrends.ca/bc-birth-tally-back-up-to-pre-pandemic-level-in-2024-2025/.

ISRCTN Registry (2024). *Assessing AI-enhanced portable ultrasound screening for developmental dysplasia of the hip in low-resource settings: a multi-center randomized trial*. DOI: https://doi.org/10.1186/ISRCTN33985096.

Jaremko, J. L. et al. (2014). "Potential for change in US diagnosis of hip dysplasia solely caused by changes in probe orientation: patterns of alpha-angle variation revealed by using three-dimensional US". In: *Radiology* 273.3, pp. 870–878. DOI: 10.1148/radiol.14140451.

Jaremko, J. L. et al. (2023). "AI aided workflow for hip dysplasia screening using ultrasound in primary care clinics". In: *Scientific Reports* 13, p. 9224. DOI: 10.1038/s41598-023-35603-9.

Johari, Ashok N. et al. (2023). "Causes of late-presenting developmental dislocation of the hip beyond 12 months of age: A pilot study". In: *Indian Journal of Medical Research* 157.5, pp. 403–411. DOI: 10.4103/ijmr.IJMR_3569_20.

Kato, H. and M. Billinghurst (1999). "Marker tracking and hmd calibration for a video-based augmented reality conferencing system". In: *Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99)*, pp. 85–94. DOI: 10.1109/IWAR.1999.803809.

Kelley, S. P. et al. (2019). "Expert-based consensus on the principles of Pavlik harness treatment for developmental dysplasia of the hip". In: *JBJS Open Access* 4.4, e18.00054. DOI: 10.2106/JBJS.OA.18.00054.

Kim, B. K., H. Tanaka, and Y. Sumi (2015). "Robotic wheelchair using a high accuracy visual marker LentiBar and its application to door crossing navigation". In: *2015 IEEE*

International Conference on Robotics and Automation (ICRA), pp. 4478–4483. DOI: 10.1109/ICRA.2015.7139819.

Koivukangas, T., J. Katisko, and J. P. Koivukangas (2013). "Technical accuracy of optical and the electromagnetic tracking systems". In: *SpringerPlus* 2.1. DOI: 10.1186/2193-1801-2-90.

Konijnendijk, A. et al. (2023). "Association between timing and duration of breech presentation during pregnancy and developmental dysplasia of the hip: A case-control study". In: *J Child Health Care* 27.1, pp. 35–45. DOI: 10.1177/13674935211042198.

Kraus, T. et al. (2024). "Universal hip dysplasia screening in Austria: what learned?" In: *Exploration of Musculoskeletal Diseases*. DOI: 10.37349/emd.2024.00049. URL: https://www.explorationpub.com/Journals/emd/Article/100749.

Krysta, Wojciech et al. (2024). "Screening of Developmental Dysplasia of the Hip in Europe: A Systematic Review". In: *Children* 11.1, p. 97. DOI: 10.3390/children11010097.

Kural, B. et al. (2019). "Risk Factor Assessment and a Ten-Year Experience of DDH Screening in a Well-Child Population". In: *Biomed Res Int*. DOI: 10.1155/2019/7213681.

Laskaratou, E. D. et al. (2024). "Epidemiology and Screening of Developmental Dysplasia of the Hip in Europe: A Scoping Review". In: *Reports (MDPI)* 7.1, p. 10. DOI: 10.3390/reports7010010.

Lasso, Andras et al. (2014). "PLUS: Open-source toolkit for ultrasound-guided intervention systems". In: *IEEE Transactions on Biomedical Engineering* 61.10, pp. 2527–2537. DOI: 10.1109/TBME.2014.2322864.

Li, Jacqueline et al. (2021). "Development of a DDH Care Pathway for India: A Study Methodology to Guide Similar Efforts in Other Countries and for Other Conditions". In: *Indian Journal of Orthopaedics* 55.6, pp. 1549–1558. DOI: 10.1007/s43465-021-00534-y.

Loder, R. T. and E. N. Skopelja (2011). "The epidemiology and demographics of hip dysplasia". In: *ISRN Orthopedics* 2011, p. 238607. DOI: 10.5402/2011/238607.

Luoto, E. S. et al. (2025). "Universal Clinical DDH Screening Complemented with Targeted Ultrasound Is Effective in Finland". In: *J Bone Joint Surg Am* 107.7, e26. DOI: 10.2106/JBJS.24.00313.

Lynch, N. A. and P. W. Shor (2025). “Building a Theory of Distributed Systems: Work by Nancy Lynch and Collaborators”. In: *arXiv* abs/2502.20468. URL: https://arxiv.org/abs/2502.20468.

Morton, T. et al. (2024). “Use of point-of-care ultrasound in rural British Columbia: Scale, training, and barriers”. In: *Canadian Family Physician* 70.2, pp. 109–116. DOI: 10.46747/cfp.7002109.

Mostofi, E. et al. (2019). “Reliability of 2D and 3D ultrasound for infant hip dysplasia in the hands of novice users”. In: *Eur Radiol*. DOI: 10.1007/s00330-018-5699-1.

NDI (2019). *Aurora Technical Specifications*. URL: https://www.ndieurope.com/wp-content/uploads/2019/11/8300163_rev009_Aurora.pdf.

— (2022). *Polaris Vega XT*. URL: https://www.ndigital.com/wp-content/uploads/2022/05/10005148-Rev004-Vega-XT.pdf.

— (2023a). *Aurora - NDI’s premier electromagnetic tracking solution*. URL: https://www.ndigital.com/electromagnetic-tracking-technology/aurora/.

— (2023b). *Electromagnetic Tracking Technology*. URL: https://www.ndigital.com/electromagnetic-tracking-technology/.

— (2024). *Polaris Vega ST - Accurate reliable optical navigation*. URL: https://www.ndigital.com/optical-navigation-technology/polaris-vega-st/.

— (n.d.). *Optotrak Certus*. URL: https://www.ndieurope.com/wp-content/uploads/2019/11/8300324_rev001_optotrak_Certus.pdf.

Olson, E. (2011). “AprilTag: A robust and flexible visual fiducial system”. In: *2011 IEEE International Conference on Robotics and Automation*. IEEE, pp. 3400–3407. DOI: 10.1109/ICRA.2011.5979561.

Open CV Documentation (2024a). *Camera Calibration and 3D Reconstruction*. URL: https://docs.opencv.org/4.x/d9/d0c/group__calib3d.html.

OpenCV Development Team (n.d.). *Detection of ArUco Markers*. URL: https://docs.opencv.org/4.x/d5/dae/tutorial_aruco_detection.html.

OpenCV Documentation (2024b). *Perspective-n-Point (PnP) pose computation*. URL: https://docs.opencv.org/4.x/d5/d1f/calib3d_solvePnP.html.

Oščádal, P. et al. (2020). “Improved Pose Estimation of Aruco Tags Using a Novel 3D Placement Strategy”. In: *Sensors* 20.17, p. 4825. DOI: 10.3390/s20174825.

Ott, J. (2024). *How much does an ultrasound machine cost?* URL: https://www.cassling.com/blog/how-much-does-an-ultrasound-machine-cost.

Pap, K. et al. (2006). "The incidence of avascular necrosis of the healthy, contralateral femoral head at the end of successful Pavlik harness in unilateral hip dysplasia". In: *International Orthopaedics* 30.3, pp. 168–170. DOI: 10.1007/s00264-006-0113-z.

Peace, J. B. and E. T. Psota (2021). "E2ETag: An end-to-end trainable method for generating and detecting fiducial markers". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Peng, C. et al. (2022). "Recent advances in tracking devices for biomedical ultrasound imaging applications". In: *Micromachines* 13.11, p. 1855. DOI: 10.3390/mi13111855.

Piwowarczyk, J., K. Mulpuri, and A. J. Hodgson (2025). "Accurate Localization of 2D Ultrasound Probes Using Inside-Out Tracking When Scanning the Neonatal Hip". In: *Computer Assisted Orthopaedic Surgery*.

PlusLib Developers (n.d.). *Ultrasound Image Orientation*. URL: https://pluslib.readthedocs.io/en/latest/UltrasoundImageOrientation.html.

Poroykov, A. et al. (2020). "Modeling ArUco Markers Images for Accuracy Analysis of Their 3D Pose Estimation". In: vol. 2744, short14–1. DOI: 10.51130/graphicon-2020-2-4-14.

Quader, N. et al. (2021). "3-D Ultrasound Imaging Reliability of Measuring Dysplasia Metrics in Infants". In: *Ultrasound in Medicine and Biology* 47.1, pp. 139–153. DOI: 10.1016/j.ultrasmedbio.2020.08.008.

Radiologyassistant.nl (2018). *Developmental Dysplasia of the Hip - Ultrasound*. URL: https://radiologyassistant.nl/pediatrics/hip/developmental-dysplasia-of-the-hip-ultrasound.

Roovers, E. A. et al. (2005). "Effectiveness of ultrasound screening for developmental dysplasia of the hip". In: *Arch Dis Child Fetal Neonatal Ed* 90.1, F25–30. DOI: 10.1136/adc.2003.029496.

Scaramuzza, D. (n.d.). *Lecture 07: Epipolar Geometry and Stereo*. URL: https://rpg.ifi.uzh.ch/docs/teaching/2015/07_multiple_view_geometry_1.pdf.

Schmidt, J. et al. (2009). "Precision, repeatability and accuracy of Optotrak® optical motion tracking systems". In: *International Journal of Experimental and Computa-*

tional Biomechanics 1, p. 114. URL: https://api.semanticscholar.org/CorpusID:96476704.

Schwend, R. M., B. A. Shaw, and L. S. Segal (2014). "Evaluation and treatment of developmental hip dysplasia in the newborn and infant". In: *Pediatric Clinics of North America* 61.6, pp. 1095–1107. DOI: 10.1016/j.pcl.2014.08.008.

Smith, J. A. (2020). "Assessing parent and healthcare professional experiences of developmental dysplasia of the hip screening". PhD thesis. University of British Columbia. URL: https://open.library.ubc.ca/media/stream/pdf/42591/1.0432065/5.

Starr, V. and B. Y. Ha (2014). "Imaging update on developmental dysplasia of the hip with the role of MRI". In: *American Journal of Roentgenology* 203.6, pp. 1324–1335. DOI: 10.2214/AJR.13.12449.

Sulaiman, A. et al. (2011). "Developmental dysplasia of hip screening using ortolani and barlow testing on breech delivered neonates". In: *Malaysian Orthopaedic Journal* 5.3, pp. 13–16. DOI: 10.5704/MOJ.1111.008.

Szentandrasi, I. et al. (2012). "Uniform marker fields: Camera localization by orientable de bruijn tori". In: pp. 319–320.

Tao, Z. et al. (2023). "Prevalence of developmental dysplasia of the hip (DDH) in infants: a systematic review and meta-analysis". In: *BMJ Paediatrics Open* 7.1, e002080. DOI: 10.1136/bmjpo-2023-002080.

Technology, Claron (2023). *MicronTracker 4*. URL: https://www.claronav.com/oem/microntracker-4.

Tirta, M. et al. (2025). "Risk Factors for Developmental Dysplasia of the Hip Before 3 Months of Age: A Meta-Analysis". In: *JAMA Network Open* 8.1, e2456153. DOI: 10.1001/jamanetworkopen.2024.56153.

Tiruveedhula, M., I. C. Reading, and N. M. P. Clarke (2015). "Failed Pavlik harness treatment for DDH as a risk factor for avascular necrosis". In: *Journal of Pediatric Orthopaedics* 35.2, pp. 140–143. DOI: 10.1097/BPO.0000000000000236.

Tsedendamba, N. et al. (2025). "Review of linear-array-transducer based volumetric ultrasound imaging techniques and their biomedical applications". In: *Bioengineering* 12.9, p. 906. DOI: 10.3390/bioengineering12090906.

Türközü, T., A. Öztürk, and A. Aydın (2020). “The evaluation of the treatment outcomes with Pavlik harness in developmental dysplasia of the hip”. In: *Eklem Hastalık Cerrahisi* 31.4, pp. 500–505. DOI: 10.5505/ejm.2020.96729.

Wang, J. and E. Olson (2016). “AprilTag 2: Efficient and robust fiducial detection”. In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 4193–4198. DOI: 10.1109/IROS.2016.7759617.

Watanabe, Y., T. Kato, and M. Ishikawa (2017). “Extended Dot Cluster Marker for High-speed 3D Tracking in Dynamic Projection Mapping”. In: *22017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. Vol. 3, pp. 52–61. DOI: 10.1109/ISMAR.2017.22.

Wicker, S. B. and S. Kim (2003). *Fundamentals of codes, graphs, and iterative decoding*. Kluwer Academic Publishers.

Woodacre, T. et al. (2014). “The costs of late detection of developmental dysplasia of the hip”. In: *Journal of Children’s Orthopaedics* 8.4, pp. 325–332. DOI: 10.1007/s11832-014-0599-7.

Xie, X. et al. (2023). “Image-guided navigation system for minimally invasive total hip arthroplasty (MITHA) using an improved position-sensing marker”. In: *International Journal of Computer Assisted Radiology and Surgery* 18.12, pp. 2155–2166. DOI: 10.1007/s11548-023-02861-x.

Xu, G. W. et al. (2020). “Ultrasound features predicting the 3-week outcome of Pavlik harness treatment for developmental hip dysplasia”. In: *Annals of Palliative Medicine* 9.3, pp. 1020–1029. DOI: 10.21037/apm-20-889.

Yaldiz, M. B. et al. (Aug. 2021). “DeepFormableTag: End-to-end Generation and Recognition of Deformable Fiducial Markers”. In: *ACM Transactions on Graphics (Proc. SIGGRAPH 2021)* 40.4. DOI: 10.1145/3450626.3459762.

Zhang, Z. (2000). “A flexible new technique for camera calibration”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.11, pp. 1330–1334. DOI: 10.1109/34.888718.

Zhu, M. et al. (2023). “HydraMarker: Efficient, Flexible, and Multifold Marker Field Generation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. DOI: 10.1109/TPAMI.2023.3287511.

Zusman, Natalie L., Pablo G. Castañeda, and Rachel Y. Goldstein (2024). “Globally inconsistent: Countries with top health indices erratic developmental hip dysplasia screening protocols”. In: *Journal of Children’s Orthopaedics* 18.4, pp. 393–398. DOI: 10.1177/18632521241229978.