Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Reproducible Simulation Benchmark for Perception-Informed Shared Control in OCT-Guided Retinal Microrobotics

Abdul Manan
Superior University
Muhammad Azam
Superior University
Nadeem Ahmad
Superior University
Waris Ali
Beijing University of Technology
Fahad Sabah
Beijing University of Technology
Raheem Sarwar
r.sarwar@mmu.ac.uk

Manchester Metropolitan University

Article

Keywords: retinal microrobotics, shared control, OCT, digital twin, safety-aware planning, uncertainty-aware control, reproducible benchmarking

Posted Date: June 17th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9731703/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A Reproducible Simulation Benchmark for Perception-Informed Shared Control in OCT-Guided Retinal Microrobotics

Abdul Manan[1], Muhammad Azam[1], Nadeem Ahmad[1], Waris Ali[2], Fahad Sabah[2], Raheem Sarwar[3*]

[1]Faculty of CS&IT, Superior University, Lahore, 54000, Pakistan.
[2]College of Computer Science, Beijing University of Technology, Beijing, 100124, China.
[3]OTEHM, Manchester Metropolitan University, Manchester, 10587, United Kingdom.

*Corresponding author(s). E-mail(s): r.sarwar@mmu.ac.uk;
Contributing authors: abdul.manan@superior.edu.pk; pmit@superior.edu.pk;
nadeemahmadap@gmail.com; warisali@emails.bjut.edu.cn; fahad.sabah@bjut.edu.cn;

Abstract

We present a reproducible simulation benchmark for evaluating perception-informed control strategies in OCT-guided retinal microrobotics. The framework links OCT-derived perception priors to safety-aware control within a two-dimensional retinal micromanipulation environment. It integrates an eight-class OCT perception module with uncertainty estimation and saliency-based target/risk map extraction, a simulated retinal workspace with synthetic deformation, remote-center-of-motion constraints, disturbances, and force-risk proxy sensing, and five interpretable controllers: naive, visual_servo, uncertainty_aware, shared, and deformation_aware. In the current benchmark run of 13,500 episodes, task success was high across controllers, with success rates ranging from 0.954 to 1.000. However, safety and trajectory-quality proxies revealed meaningful trade-offs. The naive controller achieved the lowest mean unsafe incursions and shortest final distance among several baselines, while the deformation_aware controller maintained relatively low unsafe incursions but showed reduced success and longer paths. The shared, visual_servo, and uncertainty_aware controllers achieved high success but incurred higher safety-risk proxy values. No RCM violations were observed in this benchmark configuration. The OCT perception module achieved high performance on the available static OCT dataset, while robustness transforms produced measurable degradation, supporting the inclusion of uncertainty-aware evaluation. Overall, the proposed benchmark provides a transparent and reproducible research instrument for comparing control strategies in OCT-guided retinal microrobotics. The results should be interpreted as simulation-based evidence for method development rather than clinical validation.

Keywords: retinal microrobotics, shared control, OCT, digital twin, safety-aware planning, uncertainty-aware control, reproducible benchmarking

1 Introduction

1.1 Motivation

Retinal microsurgery is an important application domain for robotic assistance because many vitreoretinal procedures require manipulation at spatial scales close to the limits of unaided human dexterity and perception. Tasks such as epiretinal membrane peeling, subretinal injection, and retinal vein cannulation involve confined intraocular workspaces, fragile tissue, small anatomical targets, and strict safety constraints [1–3]. In this setting, physiological tremor, limited depth perception, tool–tissue interaction uncertainty, and delayed corrective action may compromise surgical precision. Robotic platforms and robot-assisted instruments have therefore been explored as a means of improving motion scaling, tremor attenuation, instrument stability, and repeatability during intraocular manipulation [1, 3, 4].

Optical coherence tomography (OCT) is particularly relevant to robotic retinal surgery because it provides non-contact, depth-resolved structural imaging at micrometer-scale resolution [5, 6]. OCT has become

central to ophthalmic diagnosis and has also motivated intraoperative and instrument-integrated guidance approaches for microsurgical tasks [7, 8]. In principle, OCT-derived information can support tissue-layer localization, target identification, risk-zone estimation, and closed-loop control of surgical tools. However, most publicly available OCT datasets are designed primarily for diagnostic classification or segmentation and usually consist of static B-scans or volumetric images rather than synchronized robotic trajectories, tool poses, force measurements, operator commands, or intraoperative OCT time series [9]. This creates a methodological gap between OCT-based perception research and the development of safety-aware control strategies for OCT-guided retinal microrobotics.

A reproducible simulation benchmark can help address this gap by providing a controlled environment in which perception-derived information is connected directly to robotic control. Such a benchmark cannot replace hardware experiments or clinical validation, but it can provide an intermediate research layer for early-stage algorithm development. In particular, it enables systematic stress testing across variations in target difficulty, perceptual uncertainty, workspace geometry, tissue deformation, disturbance, and safety constraints. This is especially useful in retinal microrobotics, where large-scale physical experimentation is difficult, expensive, and often infeasible at the exploratory stage.

1.2 Scientific and Practical Relevance

The scientific motivation for this work is to evaluate OCT-derived perception outputs not only as image-level predictions, but also as control-relevant task-space signals. In many image-guided robotics pipelines, perception and control are assessed separately: perception modules are evaluated using classification, segmentation, or localization metrics, while controllers are evaluated using geometric tracking or trajectory metrics. This separation can obscure whether improved image-level performance translates into safer or more effective robotic behavior. A perception model may achieve high diagnostic accuracy while still producing poorly calibrated confidence estimates or spatially misleading saliency under image perturbation, distribution shift, or local ambiguity [10, 11]. Conversely, a controller that performs well under idealized perception may degrade when uncertainty, deformation, or risk regions are introduced.

By coupling perception priors to control, the proposed benchmark enables analysis of how uncertainty and saliency affect downstream behavior. Bayesian and approximate uncertainty-estimation methods have shown that predictive confidence can be treated as an informative signal for decision-making under model uncertainty [11, 12]. Similarly, saliency-based methods such as Grad-CAM provide spatial explanations of image-model decisions and can be used to identify image regions that influence classification outcomes [13]. In the proposed framework, such outputs are converted into target and risk priors that influence simulated robotic motion. This allows the benchmark to examine not only whether a controller reaches a target, but also whether it avoids risk regions, maintains trajectory efficiency, respects workspace constraints, and remains robust under controlled perturbations.

The practical relevance of this approach is that it supports reproducible controller development before expensive physical validation. Shared-control and virtual-fixture methods have long been used in cooperative robotic manipulation to constrain or guide user motion while preserving operator involvement [14]. In retinal microrobotics, similar principles are relevant because the controller must balance target reaching, safety margins, operator intent, and geometric constraints such as remote-center-of-motion behavior at the scleral entry point [1, 15]. A simulation benchmark with matched scenarios, deterministic seeding, and automated statistical reporting can therefore help identify promising controller designs, expose failure modes, and prioritize methods for subsequent hardware evaluation.

1.3 Gap in Existing Practice

Despite progress in retinal imaging, surgical robotics, and image-guided control, reproducible benchmarking for OCT-guided retinal microrobotic control remains limited. Public OCT datasets generally do not provide the multimodal signals required for closed-loop robotic evaluation, such as synchronized tool trajectories, force feedback, tissue deformation measurements, and intraoperative control commands [9]. At the same time, many robotic simulation studies focus on geometric reaching or path-following tasks without explicitly incorporating OCT-derived uncertainty, saliency-based risk maps, or perception-to-control coupling. As a result, it remains difficult to evaluate how image-derived uncertainty affects downstream safety and trajectory behavior.

Another limitation is the lack of standardized scenario generation and statistical reporting. Controller comparisons can be difficult to interpret when different methods are tested under different random initializations, deformation fields, disturbances, or target configurations. Aggregate means alone may also hide important safety–performance trade-offs. For example, two controllers may achieve similar success rates while differing substantially in unsafe incursions, final distance, path length, or sensitivity to deformation. Reproducibility and benchmarking have been identified as major challenges in medical robotics, where

ethical, technical, and practical constraints often limit direct replication [16]. This motivates benchmark designs that support deterministic seeding, cross-controller scenario parity, confidence interval reporting, and pairwise statistical comparisons.

The present work addresses these issues by implementing a perception-to-control benchmark with deterministic condition-level seeding and matched scenario execution across controllers. Each controller is evaluated under comparable deformation, uncertainty, and difficulty regimes, allowing observed differences to be attributed more directly to controller behavior rather than uncontrolled variation in task setup. The framework also exports episode-level and step-level rollout data, enabling detailed analysis of trajectories, safety-risk proxies, and controller responses over time.

1.4 Core Idea

The core idea of this study is to formulate OCT-guided retinal microrobotic control as a reproducible simulation proxy in which image-derived priors influence control decisions. The framework does not attempt to reproduce the full biomechanical complexity of the eye, nor does it claim clinical validity. Instead, it defines a simplified but structured two-dimensional retinal workspace in which perception-informed controllers can be evaluated under explicit constraints and perturbations. This design allows the benchmark to focus on algorithmic questions: how controllers use perception uncertainty, how they respond to deformation and disturbance, and how they trade task success against safety-risk and trajectory-quality measures.

The perception component provides an eight-class OCT module and derives uncertainty and saliency-based target/risk representations from static OCT data. These outputs are mapped into the simulated control task as priors over target regions and risk regions. The control environment includes synthetic deformation, disturbances, remote-center-of-motion constraints, and force-risk proxy sensing. Within this environment, five interpretable controller families are evaluated: `naive`, `visual_servo`, `uncertainty_aware`, `shared`, and `deformation_aware`. These controllers are intentionally interpretable rather than black-box, so that performance differences can be analyzed in relation to specific control assumptions.

This formulation is digital-twin-inspired in the sense that it links perception, environment state, constraints, and controller behavior in a unified simulation loop. However, the model is intentionally described as a simulation proxy rather than a clinically validated digital twin. Recent discussions of surgical digital twins emphasize the importance of model fidelity, validation, uncertainty, and clinical integration before digital twins can be interpreted as patient-specific decision-support systems [17, 18]. Accordingly, the present benchmark is positioned as a research instrument for method development and reproducible comparison, not as evidence of surgical safety or clinical efficacy.

1.5 Contributions

This work makes the following contributions:

- We introduce a reproducible simulation benchmark for OCT-guided retinal microrobotics that links image-derived perception priors to downstream control behavior. The benchmark is explicitly scoped as a non-clinical research instrument for algorithm development and controller comparison.
- We implement an OCT perception module that performs eight-class classification and generates uncertainty and saliency-based target/risk information. These perception-derived outputs are incorporated into the control task, allowing image-level uncertainty to influence simulated robotic behavior.
- We define a two-dimensional retinal micromanipulation workspace with synthetic deformation, disturbance, remote-center-of-motion constraints, and force-risk proxy sensing. This environment provides a structured testbed for studying safety-aware control under controlled perturbation regimes.
- We evaluate five interpretable controller families: `naive`, `visual_servo`, `uncertainty_aware`, `shared`, and `deformation_aware`. These controllers expose different assumptions about target tracking, uncertainty handling, shared autonomy, and deformation compensation.
- We provide deterministic condition-level seeding and cross-controller scenario parity. This ensures that different controllers are evaluated under matched task configurations, improving fairness and reproducibility of controller comparisons.
- We include confidence interval-based summaries, pairwise statistical comparisons, and exportable rollout data. These outputs support transparent analysis of safety–performance trade-offs beyond aggregate mean values.
- We provide mathematical formalization, algorithmic procedures, and computational considerations for the core perception-to-control pipeline. This supports reproducibility and clarifies the assumptions, scope, and limitations of the benchmark.

Overall, the proposed benchmark provides a reproducible platform for developing and comparing perception-informed control strategies in OCT-guided retinal microrobotics. The results are intended to

demonstrate the utility of the benchmark for method development, and controller analysis. They should not be interpreted as evidence of clinical safety or surgical efficacy. Rather, the framework offers a transparent simulation proxy for studying how OCT-derived perception uncertainty and risk information can be integrated into safety-aware robotic control.

2 Related Work

2.1 OCT-Based Retinal Image Analysis

Optical coherence tomography (OCT) has become one of the central imaging modalities in ophthalmology because it provides non-invasive, depth-resolved visualization of retinal microstructure at micrometer-scale resolution [5, 6]. This capability has supported extensive research on automated retinal disease classification, retinal-layer segmentation, biomarker extraction, and treatment-monitoring systems. Early large-scale deep learning studies demonstrated that convolutional neural networks can achieve strong diagnostic performance on curated OCT datasets for retinal diseases such as age-related macular degeneration, diabetic macular edema, choroidal neovascularization, and normal retinal appearance [9]. More recent work continues to show that modern deep learning architectures can reach high classification accuracy on OCT images, while also emphasizing the sensitivity of model performance to data scarcity, label noise, acquisition artifacts, and dataset bias [19].

Despite these advances, most OCT image-analysis studies treat perception as the final endpoint. In such settings, the main evaluation criteria are typically image-level accuracy, area under the receiver operating characteristic curve, segmentation overlap, or lesion-localization performance. These metrics are important for diagnostic decision support, but they do not directly measure whether OCT-derived information improves downstream robotic behavior. In OCT-guided retinal microrobotics, perception is only one component of a larger closed-loop system. A model may achieve high classification accuracy while still producing uncertain, poorly calibrated, or spatially misleading outputs under perturbation. This distinction is important because robotic control decisions depend not only on the predicted class, but also on the spatial structure, confidence, and reliability of the perception signal.

Recent developments in three-dimensional and multimodal OCT modeling further highlight the potential of richer retinal representations. Foundation-model and multimodal approaches are beginning to exploit volumetric OCT structure and complementary en face imaging modalities for diagnostic and prognostic tasks [20]. These advances suggest that OCT-derived representations may become increasingly informative for image-guided robotics. However, public OCT datasets still rarely provide synchronized robot kinematics, tool-tip trajectories, force measurements, intraoperative OCT streams, or operator-control signals. Therefore, there remains a gap between high-performing OCT perception models and reproducible evaluation of perception-informed control.

Uncertainty estimation and interpretability are also highly relevant to this problem. Calibration studies have shown that modern neural networks can be overconfident, even when their predictive accuracy is high [10]. Bayesian approximation methods and uncertainty-aware computer vision models provide mechanisms for estimating epistemic and aleatoric uncertainty, which can be useful when perception outputs are used in safety-critical decision-making [11, 12]. Similarly, saliency and class-activation methods such as Grad-CAM can identify image regions that influence model decisions, offering a way to convert image-model responses into spatial target or risk priors [13]. In the present work, OCT classification performance is therefore not interpreted as a clinical endpoint. Instead, the OCT module is used as an upstream provider of confidence, uncertainty, and saliency-derived target/risk information for downstream control evaluation.

2.2 Shared Control and Safety-Constrained Robotics

Shared control has a long history in robotic manipulation, teleoperation, and robot-assisted surgery. The central idea is to combine human or high-level task intent with robotic assistance that improves precision, stability, safety, or constraint satisfaction. In surgical robotics, shared-control methods are particularly relevant because fully autonomous execution is often neither clinically desirable nor technically mature, while purely manual operation may be limited by tremor, fatigue, depth-perception constraints, and the small scale of the operative workspace [1, 3]. Retinal microsurgery is a representative example of this challenge because it requires precise tool motion near fragile tissue, often at spatial scales close to or below normal human sensorimotor limits.

Virtual fixtures and active constraints are among the most established approaches for safety-aware shared control. These methods guide, restrict, or filter motion commands so that the tool remains within safe or task-relevant regions [14]. In minimally invasive and intraocular surgery, geometric constraints are especially important because tools enter the body through a constrained port or scleral entry site. Remote-center-of-motion (RCM) constraints are widely used to reduce lateral stress at the entry point and are central to many

minimally invasive robotic systems [15, 21]. In retinal microrobotics, the same principle is relevant because uncontrolled tool motion at the scleral entry point may increase tissue loading or compromise instrument stability.

Robotic systems for vitreoretinal surgery have demonstrated the feasibility of improving microsurgical precision through robotic assistance, motion scaling, force sensing, and image guidance. Prior work has reported robotic assistance for retinal vein cannulation, membrane manipulation, and subretinal injection tasks [2–4]. Recent reviews have further emphasized that vitreoretinal robotics is moving toward systems that combine robotic precision, intraoperative imaging, and task-specific guidance [22, 23]. For example, iOCT-guided simulated subretinal injection studies and autonomous retinal microsurgery work using RGB-D perception illustrate the growing interest in coupling visual information to robotic manipulation [24, 25].

However, safety-aware control in retinal microrobotics remains difficult to evaluate reproducibly. Hardware experiments are expensive, specialized, and limited in scale, while clinical datasets rarely contain the synchronized perception-control information needed for controlled algorithm comparison. Moreover, success alone is insufficient as a performance measure. A controller may reach the target while taking a long path, entering high-risk regions, accumulating excessive risk exposure, or relying on unrealistic assumptions about perception quality. For this reason, the present benchmark evaluates multiple proxy metrics, including task success, final distance, path length, unsafe incursions, cumulative risk, force-risk proxy behavior, and RCM violations. These metrics are not intended to replace clinical safety endpoints; rather, they provide interpretable quantities for comparing controller behavior under matched simulation conditions.

2.3 Simulation and Digital Twin Methodology

Simulation has become an important tool in medical robotics because it allows algorithms to be evaluated under controlled conditions before hardware or clinical experimentation. This is particularly valuable in domains such as retinal microrobotics, where real-world experimentation requires specialized robotic platforms, imaging systems, surgical expertise, and strict safety controls. Simulation allows perturbations, workspace geometries, deformation fields, and uncertainty regimes to be varied systematically, making it possible to study controller behavior across conditions that would be difficult to reproduce experimentally.

The term digital twin is increasingly used in medicine and surgery, but it should be applied carefully. Recent perspectives describe digital twin-assisted surgery as a future framework in which patient-specific virtual models are updated using multimodal data to support planning, guidance, prediction, and intraoperative decision-making [18]. Related surgical data-science work has also argued that digital twins require geometric scene understanding, model updating, uncertainty representation, and clinical integration before they can function as patient-specific surgical decision-support systems [17]. Under this stricter interpretation, a true surgical digital twin requires substantially more validation and patient-specific fidelity than a generic simulation environment.

For this reason, the present work uses the term simulation proxy rather than claiming a clinically validated digital twin. The proposed environment captures selected functional aspects of OCT-guided retinal micromanipulation: a retinal task space, target and risk regions, synthetic deformation, disturbances, RCM constraints, and force-risk proxy sensing. These components are intentionally simplified. They are not intended to reproduce full ocular biomechanics or patient-specific tissue behavior. Instead, they provide controlled variables for evaluating how perception-informed controllers respond to uncertainty, deformation, and safety constraints.

This distinction is consistent with broader simulation-validation literature, which emphasizes that simulation fidelity should be judged in relation to the task being studied rather than visual realism alone [26]. For algorithmic benchmarking, functional fidelity may be more important than anatomical realism: the environment must reproduce the constraints and perturbations that matter for the control question, even if it does not reproduce every physical detail of the surgical scene. In this study, deformation, force, and temporal behavior are therefore explicitly framed as proxy variables.

Reproducibility is also a key methodological concern. Surgical robotics research faces reproducibility challenges due to specialized hardware, limited access to clinical data, safety constraints, and incomplete reporting of experimental protocols [16]. A benchmark that provides deterministic condition-level seeding, cross-controller scenario parity, automated episode export, confidence intervals, and pairwise statistical comparisons can help address these limitations. The novelty of the present work is therefore positioned not as clinical biomechanical fidelity, but as the integration and reproducibility of an OCT-to-control benchmarking pipeline for early-stage method development.

Table 1 Comparison of representative related work streams and the positioning of the proposed benchmark.

Research stream	Representative focus	Typical strengths	Remaining limitations relative to this work
OCT-based retinal image analysis	Diagnostic classification, segmentation, biomarker extraction, and retinal disease prediction from OCT images or volumes [9, 27, 28]	Strong image-level performance; publicly available datasets; clinically meaningful disease labels; useful perception baselines	Usually treats perception as the final endpoint; limited or no connection to robot motion, tool trajectories, force-risk behavior, RCM constraints, or controller-level safety metrics.
OCT and retinal foundation models	Large-scale volumetric, multimodal, or foundation-model learning for OCT and retinal imaging [29, 30]	Improved generalization, representation learning, cross-device evaluation, and multimodal retinal modeling	Primarily evaluated using diagnostic or prognostic endpoints; does not directly test how uncertainty or saliency changes robotic control behavior under deformation and safety constraints.
Robot-assisted retinal microsurgery	Robotic platforms, motion scaling, tremor suppression, force sensing, and intraocular manipulation [2–4]	Demonstrates practical feasibility of robotic assistance in delicate retinal tasks; closer to surgical implementation than pure image-analysis studies	Hardware experiments are difficult to reproduce at scale; datasets and platforms are often not openly available; limited standardized benchmarking across matched uncertainty, deformation, and controller regimes.
Image-guided or autonomous retinal manipulation	Integration of visual/OCT/RGB-D perception with robotic navigation or subretinal injection workflows [24, 25]	Directly connects perception to robotic action; demonstrates the value of image guidance for retinal micromanipulation	Often task- and platform-specific; evaluation may be limited by sample size, hardware availability, or ex-vivo conditions; not primarily designed as a reusable benchmark for comparing multiple controller families.
Shared control and virtual fixtures	Cooperative control, active constraints, motion filtering, and safety-aware guidance [1, 14]	Provides interpretable mechanisms for combining operator intent, task constraints, and robotic assistance	Often evaluated in generic manipulation or platform-specific surgical settings; does not necessarily incorporate OCT-derived uncertainty, saliency-based risk maps, or retinal deformation proxies.
Surgical simulation and digital twins	Simulation environments, patient-specific models, and digital-twin concepts for planning, training, or intraoperative guidance [17, 18]	Supports controlled experimentation, model-based prediction, and future patient-specific surgical decision support	Clinically meaningful digital twins require substantial validation and patient-specific fidelity; many simulations do not provide reproducible OCT-to-control pipelines or controller-level statistical comparisons.
Reproducible surgical robotics methodology	Benchmarking, replication, experimental reporting, and reproducibility in medical robotics [16]	Highlights the importance of deterministic protocols, transparent reporting, shared data, and standardized evaluation	Provides methodological motivation but not a domain-specific OCT-guided retinal microrobotics benchmark.

2.4 Limitations of Existing Work and Positioning of the Present Benchmark

The proposed benchmark is positioned relative to three related but partially separate research streams: OCT-based retinal image analysis, robotic/shared-control methods for retinal micromanipulation, and simulation or digital-twin methodology in medical robotics. Each stream has produced important advances, but none directly provides a reproducible OCT-to-control benchmark in which perception uncertainty, saliency-derived risk information, deformation, remote-center-of-motion constraints, and controller-level safety metrics are evaluated together.

Table 1 summarizes the main methodological focus, representative contributions, and remaining limitations of existing work. OCT-based retinal image-analysis studies have established strong diagnostic and representation-learning baselines, including large-scale deep learning models, public OCT datasets, and recent volumetric or multimodal foundation models [9, 27–30]. However, these studies usually evaluate image-level endpoints, such as classification accuracy, segmentation performance, or disease-prediction metrics. They do not generally examine how OCT-derived confidence, uncertainty, or saliency affects downstream robotic motion.

Robotic retinal surgery and shared-control studies address the manipulation problem more directly. Prior work has demonstrated robot-assisted intraocular surgery, force-sensing instruments, image-guided retinal manipulation, and autonomous or semi-autonomous subretinal injection workflows [2–4, 25]. These studies are essential for demonstrating feasibility in hardware, ex-vivo, or early clinical settings. Nevertheless, they are often difficult to reproduce at scale because they require specialized robotic platforms, imaging systems, biological tissue models, and expert operators. Moreover, many robotic evaluations focus on task completion or trajectory behavior without standardized benchmarking across perception uncertainty, deformation regimes, and matched controller comparisons.

Simulation and digital-twin studies provide a third relevant direction. Surgical digital twins are increasingly discussed as a future framework for patient-specific planning, intraoperative guidance, and decision support [17, 18]. However, the digital-twin literature also emphasizes that clinically meaningful twins require

model fidelity, patient-specific data, uncertainty representation, validation, and continuous updating. Under this stricter definition, the present work should not be interpreted as a clinically validated surgical digital twin. Instead, it is more accurately described as a simulation proxy or digital-twin-inspired benchmark. Its purpose is to provide a transparent and reproducible environment for studying algorithmic behavior before hardware or clinical validation.

This positioning clarifies the contribution of the present work. The benchmark does not attempt to outperform diagnostic OCT models on clinical image-analysis tasks, nor does it replace robotic hardware studies or patient-specific digital twins. Instead, it addresses a narrower but important methodological gap: the lack of a reproducible benchmark for studying how OCT-derived perception priors influence safety-aware control behavior. In this sense, the novelty lies in the integration of perception, uncertainty, deformation, shared control, RCM constraints, and statistical reporting within a single executable pipeline.

The limitations of existing work also motivate the interpretation of the present results. High task success in simulation should not be read as evidence of clinical readiness. Similarly, unsafe incursions, cumulative risk, and force-risk estimates are proxy metrics rather than validated surgical safety endpoints. Their value is comparative: they allow different controllers to be evaluated under matched conditions and make safety–performance trade-offs visible. This is particularly important because more sophisticated controllers do not necessarily dominate simpler baselines across all metrics. A reproducible benchmark therefore provides a useful method-development stage before moving toward hardware experiments, ex-vivo validation, or clinically grounded digital-twin models.

3 Problem Formulation and Preliminaries

We consider a robotic system that must plan and execute tool-tip trajectories in the image plane of an OCT (Optical Coherence Tomography) volume, using visual perception to estimate anatomical risk and target locations. The problem is cast as a finite-horizon constrained optimal control problem under uncertainty, where the controller must balance safety (encoded by risk, tissue stiffness, and force limits) with task completion (reaching a target point).

3.1 Notation and Definitions

Image and perception. Let $I \in \mathbb{R}^{H\times W\times 3}$ denote a static pre-operative OCT image, which can be an RGB rendering of a grayscale slice. The image covers a rectangular region $\Omega \subset \mathbb{R}^2$ with pixel coordinates (i, j), $i = 1, \dots, H$, $j = 1, \dots, W$. A pre-trained perception model $f_{\text{perc}} : \mathbb{R}^{H\times W\times 3} \to \Delta^{K-1} \times [0, 1] \times [0, 1]^{H\times W}$ processes the image and outputs three quantities:

- A class posterior distribution $p \in \Delta^{K-1}$, where $\Delta^{K-1} = \{p \in [0, 1]^K \mid \sum_{k=1}^K p_k = 1\}$ is the K-dimensional probability simplex. Each component p_k models the confidence that the central region (or the whole slice) belongs to retinal layer k.
- An uncertainty scalar $u \in [0, 1]$ that quantifies the model's epistemic uncertainty about the scene (e.g., from out-of-distribution regions or low contrast).
- A saliency map $S \in [0, 1]^{H\times W}$ that highlights pixels the model deems relevant for the task (e.g., layer boundaries, lesions).

Risk and deformation maps. From the perception outputs and additional (synthetic or real) knowledge we define three spatial fields over the image domain Ω:

- **Risk map** $R : \Omega \to [0, 1]$, which assigns a scalar risk value to every pixel. In the benchmark, R is constructed by combining the saliency map S and the uncertainty u, for example $R(i, j) = u \cdot S(i, j)$ followed by thresholding or a soft weighting of anatomical priors. $R(x)$ for a continuous position $x \in \Omega$ is obtained by bilinear interpolation.
- **Deformation magnitude field** $D : \Omega \to \mathbb{R}_{\geq 0}$, representing the expected local tissue deformation per unit applied force. High values indicate compliant tissue where tool motion may cause large displacements, increasing the risk of collateral damage.
- **Force-risk proxy** $F : \Omega \to \mathbb{R}_{\geq 0}$, which encodes the danger associated with exerting force at a given location (e.g., near critical structures). Together with D, it governs the cost of tool-tissue interaction.

Target and tool state. A target point $x^\star \in \Omega$ defines the desired final tool-tip position. This target may be extracted from the saliency map (e.g., the centroid of a lesion) or provided manually. At discrete time $t = 0, 1, \dots, T - 1$, the tool-tip state consists of its position $x_t \in \Omega \subset \mathbb{R}^2$ and its current velocity $v_t \in \mathbb{R}^2$. The combined state is written as $s_t = (x_t, v_t)$.

Action and dynamics. The controller selects an action $\Delta x_t \in \mathbb{R}^2$, which represents a desired displacement increment. The actual tool-tip evolution follows the stochastic dynamics

$$x_{t+1} = x_t + \Delta x_t + \xi_t, \qquad v_{t+1} = \Delta x_t, \tag{1}$$

subject to hard bounds $x_t \in \mathcal{W} \subseteq \Omega$ (the workspace, typically the image interior with a safety margin) and $\|\Delta x_t\| \le \Delta_{\max}$ (maximum step size). Velocities are implicitly bounded through the action limits. The disturbance $\xi_t \in \mathbb{R}^2$ is a bounded random vector with $\|\xi_t\| \le \xi_{\max}$ and zero mean, representing unmodelled tissue motion or tracking errors.

Remote center of motion (RCM). In many retinal procedures the instrument must pivot around a fixed entry point (the sclerotomy). When the image plane is a projection of the 3-D workspace, the 2-D motion must respect a proxy RCM constraint: the line through the tool shaft should intersect the image plane at a fixed trocar point $x_{\text{rcm}} \in \mathbb{R}^2$. This is enforced by an optional projection operator $\text{Proj}_{\text{RCM}}(x_t, \Delta x_t)$ that maps the planned displacement to the nearest admissible displacement satisfying the RCM condition. The projected action is then used in the dynamics.

Policy and cost. The decision maker is a deterministic policy π that maps the current state s_t and the available task information (maps R, D, F, target $x^\star$, time step t) to an action Δx_t. The immediate cost function $c(x_t, \Delta x_t; R, D, F, \text{RCM})$ penalizes:

- distance to target: $\|x_t - x^\star\|^2$,
- risk at current position: $R(x_t)$,
- deformation-weighted path length: $D(x_t)\,\|\Delta x_t\|$,
- force-risk weighted effort: $F(x_t)\,\|\Delta x_t\|$,
- control energy: $\|\Delta x_t\|^2$,
- optional RCM violation penalty if the projected action deviates from the requested one.

Coefficients $w_* > 0$ balance these terms. The total trajectory cost is the sum of discounted (or undiscounted) stage costs over the horizon T.

3.2 Task Objective

For each episode—defined by a static OCT image, a fixed target $x^\star$, and generated fields R, D, F—the controller seeks a policy π that minimizes the expected cumulative cost subject to dynamics and workspace constraints:

$$\min_{\pi}\ \mathbb{E}_{\xi_0,\dots,\xi_{T-1}}\left[\sum_{t=0}^{T-1} c\big(x_t, \Delta x_t; R, D, F, \text{RCM}\big)\right] \quad \text{s.t.} \quad \begin{aligned} x_{t+1} &= f(x_t, \Delta x_t, \xi_t), \quad v_{t+1} = \Delta x_t, \\ x_t &\in \mathcal{W},\ \|\Delta x_t\| \le \Delta_{\max},\ x_0 \text{ given} \end{aligned} \tag{2}$$

The expectation is taken over the sequence of bounded disturbances ξ_t, which are i.i.d. or drawn from a fixed distribution per episode. The policy π may be reactive (closed-loop) and can exploit knowledge of the current state, the maps, and the target. The solution is a sequence of actions $\Delta x_0, \dots, \Delta x_{T-1}$ that brings the tool-tip close to $x^\star$ while keeping risk, deformation, and force penalties low.

4 Methodology

This section details the components that instantiate the abstract problem formulation of Section 3 into a concrete benchmark. We describe the construction of perception-derived risk and target priors, the digital twin dynamics with safety proxies, the family of controllers under test, and the training regime for the underlying perception model. The section concludes with the simulation loop and evaluation metrics that unite all modules.

4.1 Perception-Derived Task Priors

The perception model outputs a class distribution p, uncertainty scalar u, and saliency map S (see Section 3.1). From these we construct a spatial risk map R and a target point $x^\star$ that encode task-specific anatomical knowledge.

Risk map. The risk map $R \in [0,1]^{H \times W}$ is a weighted heuristic combination of three terms:

$$R(i,j) = \text{clip}\Big(w_s R_{\text{sal}}(i,j) + w_u R_{\text{unc}}(i,j) + w_c\, r_{\text{class}},\ 0,\ 1\Big) \tag{3}$$

where:

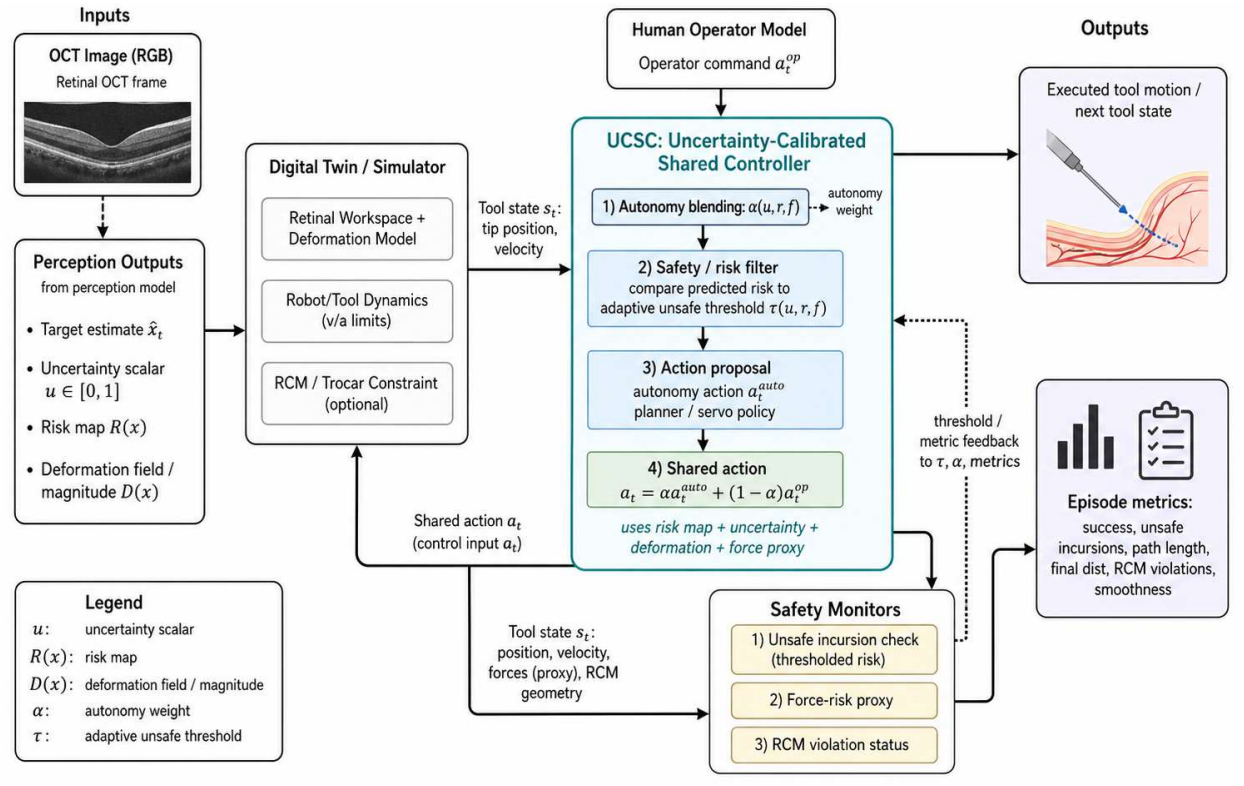


Fig. 1 System architecture of the UCSC Shared-Autonomy Controller in Retinal Microrobotics Digital Twin. The diagram illustrates the flow of inputs, the shared-autonomy controller's processing modules, safety monitors, and the resulting outputs. Key components include OCT image inputs, perception model outputs, digital twin simulation, human operator interaction, and safety-critical feedback mechanisms.

- R_{sal} is obtained from the raw saliency map S by applying a small Gaussian blur (kernel $\sigma = 2$ pixels) and normalising to the range $[0, 1]$. This emphasises regions the model attended to, which often correspond to layer boundaries or lesions.
- R_{unc} is a uniform map filled with the scalar uncertainty u. It acts as a global risk amplifier when the model is unsure about the whole scene.
- r_{class} is a scalar prior risk associated with the predicted class $\hat{k} = \arg\max p$. A hand-tuned lookup table maps each of the K classes to a base risk value (e.g., 0.1 for vitreous, 0.8 for retinal pigment epithelium, 0.5 for fluid). This injects semantic knowledge without requiring dense annotation.

The coefficients w_s, w_u, w_c (default $0.5, 0.3, 0.2$) balance local saliency, global uncertainty, and class prior. The clip$(\cdot, 0, 1)$ function ensures a valid probability range.

Target extraction. A target point $x^\star \in \Omega$ is derived from the saliency map S. First, S is thresholded at $\tau_s = 0.6$ to obtain a binary hotspot mask; the centroid of the largest connected component is taken as the primary target μ. Then, an uncertainty-adaptive acceptance radius is defined as

$$r_{\text{target}} = 6 + 26\,u \quad \text{(pixels)} \tag{4}$$

This radius serves two purposes: it defines a goal region for the controllers (distance $\leq r_{\text{target}}$ counts as success) and can be used by planners as a soft constraint. When u is high, the target region expands, relaxing the required precision.

4.2 Digital Twin Dynamics and Safety Proxies

The tool-tip motion is simulated in 2D image coordinates using a kinematic model with projection and perturbation layers.

Dynamics. Given a desired action (displacement) Δx_t, a commanded velocity is computed as $v_{t+1} = \Delta x_t / \Delta t$ (with $\Delta t = 1$ time unit). The actual next position is obtained by

$$x_{t+1} = \Pi_{\mathcal{W}}\Big(\Pi_{\text{RCM}}\big(x_t + v_{t+1}\Delta t + \epsilon_t\big)\Big) \tag{5}$$

where:

- Π_{RCM} enforces the remote centre of motion. A fixed trocar point $x_{\text{rcm}} \in \mathbb{R}^2$ is defined in the image plane. The admissible set of displacements are those that keep the tool-shaft line intersecting x_{rcm}. The projection $\Pi_{\text{RCM}}(\tilde{x})$ finds the nearest point $\tilde{x}$ such that the vector $x_t \to \tilde{x}$ is tangent to the circle centred at x_{rcm}; if the current position is already compliant, the identity is returned. This is a simplified 2-D analogue of the true RCM constraint.
- $\Pi_{\mathcal{W}}$ clips the position to the workspace $\mathcal{W}$, which is the image interior eroded by a 5-pixel margin.
- $\epsilon_t \sim \mathcal{U}([-\sigma, \sigma]^2)$ is additive bounded disturbance modelling tissue motion, registration errors, or low-level control inaccuracies. We set $\sigma = 0.5$ pixels.

The velocity state is updated as $v_{t+1} = (x_{t+1} - x_t)$ (assuming $\Delta t = 1$).

Force-risk proxy. During execution, a scalar force-risk proxy $F_t \in [0, 1]$ is computed at each step to approximate the danger associated with the current motion and tissue interaction:

$$F_t = \text{clip}\Big(0.45\, R(x_t) + 0.25\, \tilde{D}(x_t) + 0.20\, \tilde{s}_t + 0.10\, \tilde{j}_t,\ 0,\ 1\Big) \tag{6}$$

with the following normalised terms:

- $R(x_t)$ is the risk value at the tool-tip (bilinearly interpolated from R).
- $\tilde{D}(x_t) = \min(D(x_t)/D_{\max}, 1)$ is the normalised deformation magnitude, where $D_{\max} = \max_{i,j} D(i, j)$.
- $\tilde{s}_t = \min(\|v_t\|/v_{\max}, 1)$ is the normalised tool speed ($v_{\max} = 10$ pixels/step).
- $\tilde{j}_t = \min(\|v_t - v_{t-1}\|/j_{\max}, 1)$ is the normalised jerk (acceleration magnitude), with $j_{\max} = 5$ pixels/step^2.

The coefficients weight risk, tissue compliance, speed, and sudden acceleration, reflecting that high force is likely when moving fast in stiff tissue near sensitive structures. F_t is used both as an auxiliary safety metric and as a penalty term in deformation-aware planning.

Unsafe incursion detection. A step is flagged as *unsafe* if the local risk at the new position exceeds a high threshold:

$$\text{unsafe}_t = \mathbf{1}\big[R(x_{t+1}) \geq \tau_{\text{unsafe}}\big], \qquad \tau_{\text{unsafe}} = 0.7 \tag{7}$$

This binary signal is accumulated over the episode and reported as a safety metric.

4.3 Controller Family

We implement five structured controllers that cover a spectrum from naive reactive policies to optimisation-based planners. All controllers output a displacement $\Delta x_t \in \mathbb{R}^2$ given the current state and task information.

Naive controller.

A constant-step-size direction-only policy that always moves along the line toward the target:

$$\Delta x_t = \alpha\, \frac{x^\star - x_t}{\|x^\star - x_t\|_2}, \quad \alpha = 2.0 \text{ pixels} \tag{8}$$

No risk or dynamics awareness is present; it serves as an absolute baseline.

Visual servo (proportional).

A standard proportional feedback law with a saturation:

$$\Delta x_t = \text{clip}_\rho\big(k_p(x^\star - x_t)\big), \quad k_p = 0.8,\ \rho = 4.0 \text{ pixels} \tag{9}$$

$\text{clip}_\rho(\cdot)$ scales the vector magnitude to at most ρ if it exceeds the bound. This controller adapts step length when far from the target but ignores risk.

Uncertainty-aware controller.

Extends the visual servo by modulating the step size based on the global uncertainty u_t (the perception model's uncertainty output at the current image, which we treat as a scalar per episode):

$$\Delta x_t = s(u_t)\, \Delta x_t^{\text{servo}}, \quad s(u) = \text{clip}(1 - 0.85\, u,\ s_{\min},\ 1) \tag{10}$$

with $s_{\min} = 0.2$. When the model is confident ($u \approx 0$), the step is unchanged. As uncertainty grows, the step size is aggressively reduced (by up to 85%), causing slower, more cautious approach. This is a simple heuristic for risk-averse behaviour without spatial reasoning.

Algorithm 1 Short-Horizon Deformation-Aware Planning

Require: Tool-tip position x, target $x^\star$, risk map R, deformation map D, force-risk proxy F_t, uncertainty u, candidate step sizes $\mathcal{S}$, candidate directions Θ

$C^\star \leftarrow +\infty$, $a^\star \leftarrow (0, 0)$
for $s \in \mathcal{S}$ **do**
  for $\theta \in \Theta$ **do**
    $a \leftarrow s\,[\cos\theta,\ \sin\theta]^\top$
    $x' \leftarrow x + a$
    $d \leftarrow \|x' - x^\star\|_2$, $r \leftarrow R(x')$, $d_{\text{def}} \leftarrow D(x')$
    $\text{rcm_viol} \leftarrow \mathbf{1}\big[\|\Pi_{\text{RCM}}(x') - x'\| > \epsilon_{\text{rcm}}\big]$
    $C \leftarrow w_d d + w_r\, 10\, r + w_\delta\, 4\, d_{\text{def}} + w_f\, 8\, F_t + w_g\, 12\, \text{rcm_viol}$
    if $C < C^\star$ **then**
      $C^\star \leftarrow C$, $a^\star \leftarrow a$
    end if
  end for
end for
return $\Delta x_t = a^\star \cdot \text{clip}(1 - 0.6u,\ 0.3,\ 1)$

Shared autonomy controller.

Mimics human-robot collaboration. At each step, an *operator command* a_t^{op} is generated synthetically (e.g., a directional input toward the target perturbed by noise) and an *autonomy command* a_t^{au} is obtained from the visual servo. The executed action is a convex blend:

$$\tilde{a}_t = (1 - \lambda_t)\, a_t^{\text{op}} + \lambda_t\, a_t^{\text{au}} \tag{11}$$

where the mixing weight depends on uncertainty:

$$\lambda_t = \text{clip}\big(\lambda_0 + 0.3\, u_t,\ 0,\ 0.9\big), \quad \lambda_0 = 0.2 \tag{12}$$

Thus, when uncertainty is low, the operator input dominates (with a small autonomous bias); when uncertainty is high, the autonomy takes over, up to 90%. After blending, a *risk-aware filter* is applied: if the predicted next position under $\tilde{a}_t$ has $R(x_t + \tilde{a}_t) \geq \tau_{\text{unsafe}}$, the action is scaled down in magnitude until the risk falls below the threshold; if no safe scaling exists, the action is replaced by a zero step. This filter adds a local safety layer on top of the arbitration.

Deformation-aware planner.

A one-step model-predictive controller that searches over a discrete set of candidate actions and selects the one minimising a cost function. The set of candidates consists of N_s step sizes $\mathcal{S} = \{1, 2, 3, 4\}$ pixels and N_θ directions $\Theta = \{0, \pi/4, \ldots, 7\pi/4\}$, yielding up to 32 actions (plus a zero-step). For each candidate a, the next position $x' = x_t + a$ is evaluated using a cost:

$$C(x'; x^\star, R, D, F_t) = w_d\, d(x', x^\star) + w_r\, 10\, R(x') + w_\delta\, 4\, D(x') + w_f\, 8\, F_t + w_g\, 12\, \mathbf{1}_{\text{RCM-viol}}, \tag{13}$$

where $d(\cdot, \cdot)$ is Euclidean distance, $R(x')$ and $D(x')$ are interpolated maps, F_t is the current force-risk proxy (which already includes speed/jerk terms, thus penalising large steps implicitly), and the RCM violation indicator is 1 if $\|\Pi_{\text{RCM}}(x') - x'\| > \epsilon_{\text{rcm}}$. The weights are set to $w_d = 1$, $w_r = 2$, $w_\delta = 1.5$, $w_f = 2$, $w_g = 5$, chosen to prioritise safety over progress.

After the action $a^\star$ with minimal cost is found, it is further scaled by an uncertainty factor to discourage confident large steps in high-uncertainty regions:

$$\Delta x_t = a^\star \cdot \text{clip}(1 - 0.6\, u_t,\ 0.3,\ 1) \tag{14}$$

The complete procedure is formalised in Algorithm 1. This controller explicitly balances target attraction against spatial risk and deformation, and can incorporate the RCM constraint into the cost.

4.4 Training and Evaluation of the Perception Model

The perception backbone f_{perc} is a neural network (e.g., EfficientNet-B3 or a lightweight ViT) that maps an OCT slice to (p, u, S). It is trained on a separate dataset of labelled OCT images with K retinal layer/fluid classes. The training objective is weighted cross-entropy with optional label smoothing:

$$\mathcal{L} = -\sum_{k=1}^{K} \omega_k\, \tilde{y}_k \log \hat{p}_k \tag{15}$$

Algorithm 2 Benchmark Episode Execution

Require: Scenario data: maps R, D, target $x^\star$, uncertainty u, workspace $\mathcal{W}$, RCM point x_{rcm}, time horizon T, unsafe threshold τ_{unsafe}, success radius r_{target}
Ensure: Metrics: success boolean, cumulative risk, path length, force proxy integral, smoothness, RCM violations, unsafe incursions

Initialise tool position $x_0 \sim \mathcal{U}(\mathcal{W})$ (at least 20 pixels from target), $v_0 = (0, 0)$, $v_{-1} = v_0$
for $t = 0$ to $T - 1$ **do**
  Generate synthetic operator command a_t^{op} (if required by controller)
  Compute force-risk proxy F_t using (6)
  Form observation $o_t = (x_t, v_t, x^\star, R, D, u, a_t^{\text{op}}, F_t)$
  $\Delta x_t \leftarrow \pi(o_t)$
  Apply dynamics: $x_{t+1} \leftarrow \Pi_{\mathcal{W}}\big(\Pi_{\text{RCM}}(x_t + \Delta x_t + \epsilon_t)\big)$, $\epsilon_t \sim \mathcal{U}([-\sigma, \sigma]^2)$
  Update $v_{t+1} \leftarrow x_{t+1} - x_t$
  Accumulate metrics:
    d_{path} += $\|\Delta x_t\|$,
    risk_cum += $R(x_{t+1})$,
    force_cum += F_t,
    unsafe_cnt += $\mathbf{1}[R(x_{t+1}) \geq \tau_{\text{unsafe}}]$,
    rcm_cnt += $\mathbf{1}[\|\Pi_{\text{RCM}}(x_t + \Delta x_t) - (x_t + \Delta x_t)\| > \epsilon_{\text{rcm}}]$,
    smoothness += $\|v_{t+1} - v_t\|$
  if $\|x_{t+1} - x^\star\| \leq r_{\text{target}}$ **then**
    success $\leftarrow$ True; **break**
  end if
end for
return success, total metrics

where $\hat{p}_k$ is the predicted probability for class k, $\tilde{y}_k$ is the (possibly smoothed) one-hot ground truth, and ω_k are class weights set inversely proportional to class frequency in the training set to combat imbalance. Label smoothing of 0.1 is used to mitigate overconfidence.

The model is trained with the AdamW optimiser, a base learning rate of 1×10^{-3}, and a cosine annealing schedule that decays the rate to 1×10^{-5} over 100 epochs. Mixed precision (AMP) is enabled to reduce memory and speed up training. A validation set of 20% of the data is evaluated after each epoch, and the checkpoint with the highest macro-F1 score is selected (early stopping with patience of 20 epochs). The uncertainty branch is trained via an auxiliary loss that encourages low uncertainty on correctly classified in-distribution examples and high uncertainty on corrupted or out-of-distribution inputs; for simplicity, we follow the approach of and treat the scalar output as a learned confidence estimate.

During benchmark episodes, the perception model is frozen and its outputs are used solely to derive priors; no online learning or adaptation occurs.

4.5 Episode Execution and Evaluation Metrics

Each benchmark run is an *episode* consisting of a static scene $(I, x_0^\star)$ from which the maps R, D, F and the initial target $x^\star$ are precomputed. The controller π interacts with the digital twin over T time steps. Algorithm 2 outlines the standard simulation loop.

The metrics collected allow quantitative comparison across controllers:

- **Success rate:** fraction of episodes where the tool reaches the target region within T steps without becoming unsafe (a run is terminated if an unsafe incursion persists for 3 consecutive steps).
- **Cumulative risk:** sum of $R(x_t)$ over trajectory, penalising proximity to high-risk areas.
- **Path length** and **smoothness** (total jerk) measure efficiency and control quality.
- **Force-risk integral** $\sum F_t$ reflects the amount of dangerous interaction.
- **Unsafe incursions** and **RCM violations** count safety-related events.

These statistics are averaged over 100 episodes with randomised starting positions, target perturbations, and disturbance realisations to obtain robust performance estimates. The benchmark is designed such that the perception model, risk map generation, and dynamics are fully reproducible, enabling fair algorithmic comparisons.

5 Theoretical Analysis

While the benchmark is designed for empirical comparison of control strategies, several theoretical properties of its components are essential for reproducibility, computational predictability, and understanding the limits of the safety heuristics. This section formalises those properties.

5.1 Boundedness and Computational Guarantees

A fundamental prerequisite for simulation stability and real-world deployability is that all controllers output bounded actions, preventing violations of the workspace and ensuring finite evaluation time. We establish a uniform bound on the action norm for every controller in the family described in Section 4.3.

Proposition 1 (Uniform action boundedness) *For any state s_t and task prior, every controller π defined in Section 4.3 produces an action $\Delta x_t \in \mathbb{R}^2$ satisfying $\|\Delta x_t\|_2 \leq \Delta_{\max}$, where $\Delta_{\max} = 4$ pixels.*

Proof sketch We examine each controller individually:

- **Naive controller** (Eq. 8) uses a fixed step length $\alpha = 2.0$, so $\|\Delta x_t\| = 2 \leq 4$.
- **Visual servo** (Eq. 9) applies $\text{clip}_\rho(\cdot)$ with $\rho = 4.0$, directly bounding the norm by 4.
- **Uncertainty-aware** (Eq. 10) modulates the visual servo action by $s(u) \in [s_{\min}, 1] = [0.2, 1]$, thus $\|\Delta x_t\| \leq \rho = 4$.
- **Shared autonomy** blends operator and autonomy commands (max norms bounded by ρ after clipping) and then applies a risk-aware filter that clips the blended action magnitude to at most `max_step_px` (set to 4) in its fallback mode; consequently $\|\tilde{a}_t\| \leq 4$.
- **Deformation-aware planner** (Algorithm 1) selects from a finite set of candidate magnitudes $\mathcal{S} = \{1, 2, 3, 4\}$, and the final action is scaled by a factor in $[0.3, 1]$. Therefore $\|\Delta x_t\| \leq 4$.

All cases obey the same global bound. The explicit clipping and finite candidate sets guarantee that no action exceeds $\Delta_{\max}$. □

This boundedness ensures that the dynamics update $x_{t+1} = \Pi_{\mathcal{W}}(\Pi_{\text{RCM}}(x_t + \Delta x_t + \epsilon_t))$ remains well-defined and that the tool cannot exit the workspace in a single step without being projected back, supporting reliable simulation rollouts.

Computational complexity of the planner. Among the controllers, only the deformation-aware planner involves non-trivial search. The following proposition quantifies its efficiency.

Proposition 2 (Planner complexity) *The short-horizon planner (Algorithm 1) runs in time $\mathcal{O}(|\mathcal{S}|\,|\Theta|)$ and uses $\mathcal{O}(1)$ additional memory beyond the map storage.*

Proof The algorithm iterates over $s \in \mathcal{S}$ and $\theta \in \Theta$, i.e., $|\mathcal{S}| \times |\Theta|$ iterations. Inside the inner loop, evaluating x', computing distance, bilinear interpolation for R and D, checking RCM violation, and computing the cost C all require constant time. The algorithm maintains only a scalar $C^\star$ and a vector $a^\star$, i.e., constant memory overhead. □

In the benchmark instantiation, $|\mathcal{S}| = 4$, $|\Theta| = 8$, yielding at most 32 cost evaluations per step. This is negligible even for high-frequency control loops and can be trivially vectorised on a GPU if needed. The constant memory footprint guarantees that the planner can be embedded on resource-constrained platforms.

5.2 Reproducibility Guarantees

A key requirement for fair benchmarking is that identical experimental conditions produce identical scenarios and environment dynamics across different runs and machines. The benchmark achieves this via deterministic hashing from high-level condition descriptors to random seeds.

Proposition 3 (Condition-level reproducibility) *Let $\mathcal{C} = (\textit{seed}, \textit{difficulty}, \textit{deformation}, \textit{uncertainty}, \textit{episode_idx})$ be a tuple of integers (or enumerations) specifying an experimental condition. Under the implemented protocol, all random numbers used to generate the scene (risk map R, deformation map D, target perturbation $\delta x^\star$), the initial tool position x_0, and the disturbance sequence $\{\epsilon_t\}$ are deterministically derived from $\mathcal{C}$ via a stable hash function. Consequently, two independent executions with the same $\mathcal{C}$ yield identical $R, D, x^\star, x_0$, and disturbance realisations.*

Proof sketch The condition tuple is serialised to a canonical string representation (e.g., `f"seed_{difficulty}_..._{episode_idx}"`). This string is hashed using BLAKE2s with a fixed secret key, producing a 256-bit digest. The digest is then used to seed a chain of pseudorandom number generators (PRNGs) via NumPy's `SeedSequence`. Each component of the scenario (risk map augmentation, deformation field perturbation, target offset, start position, disturbance sequence) draws from an independent PRNG initialised from a sub-seed derived from the same digest. Because BLAKE2s is a deterministic, collision-resistant hash and the serialisation is unambiguous, the mapping $\mathcal{C} \mapsto$ seeds is invariant across Python versions (using stable libraries) and independent of Python's hash randomisation. Thus, any re-execution of the same $\mathcal{C}$ produces bit-identical environment states. □

This property is essential for comparative evaluations: any observed performance difference between controllers under the same condition tuple can be attributed solely to the algorithmic choices, not to randomness in scenario construction.

5.3 Limitations of the Safety Heuristic

All controllers that incorporate the risk-aware filter (shared autonomy) or the one-step cost penalty (deformation-aware planner) rely on a local, one-step lookahead of the risk map. While empirically effective at reducing unsafe incursions, this mechanism does *not* constitute a formal safety shield.

Local nature of the safety filter The risk-aware filter (used in shared autonomy) and the deformation-aware planner's cost term are local heuristics that can prevent predicted unsafe actions in the immediate next step, but they do not guarantee global safety invariance over the entire trajectory. Specifically, there exist risk map geometries, target placements, and initial states for which the tool can be forced into a state x_t with $R(x_t) \geq \tau_{\text{unsafe}}$ at a future time step, even though every single action passes the local check.

The filter considers only $R(x_t + \Delta x_t)$; it cannot foresee that a sequence of individually safe steps may lead the tool into an enclosed high-risk region with no escape path that avoids crossing τ_{unsafe}. Furthermore, the filter has no knowledge of dynamic constraints beyond the step bound, and it does not plan a recovery trajectory. In control-theoretic terms, the policy is not a control barrier function (CBF)-based safety filter; it lacks forward-invariance guarantees. This limitation is by design: the benchmark serves as a testbed where such safety deficits can be exposed and quantified, motivating the development of more sophisticated planning algorithms.

Thus, the safety metrics reported (unsafe incursions, cumulative risk) must be interpreted as empirical indicators under a specific risk map distribution, not as provable upper bounds on worst-case risk. The benchmark's value lies precisely in its ability to highlight these shortcomings under repeatable conditions.

6 Experimental Setup

We design an extensive simulation campaign to evaluate the controller family under varying perceptual uncertainty, tissue deformation, task difficulty, and geometric constraints. All procedures are implemented as a command-line pipeline with declarative YAML configuration and deterministic seeding, ensuring full reproducibility (Section 5.2).

6.1 Implementation and Pipeline

The entire experimental workflow is orchestrated by a Python command-line interface (CLI) that reads a single YAML configuration file specifying all factor levels, controller hyperparameters, and output paths. The pipeline is hardened to ensure that every stage can be independently rerun and that intermediate artifacts are hashed to avoid redundant computation. The top-level execution stages are:

1. **Benchmark generation.** Based on the factorial design, generate all scenario tuples $(R, D, x^\star)$ and seed combinations. For each condition, instantiate the digital twin, create the five controller instances, and run the prescribed number of episodes. Log raw per-step metrics, success flags, and terminal states.
2. **Perception evaluation.** On a held-out set of OCT images, compute the classification and calibration metrics of the frozen perception model, including Monte-Carlo estimates of uncertainty quality. This step may leverage a pre-computed evaluation if available.
3. **Statistical summary and paper assets.** Aggregate raw logs into pandas DataFrames, compute summary statistics (mean, median, confidence intervals), generate LaTeX tables, and produce publication-ready plots (success-risk Pareto fronts, trajectory overlays, sensitivity curves). All outputs are organised in a timestamped run directory.

The pipeline uses Hydra for configuration management, DVC for artifact tracking, and Weights & Biases (optional) for live logging. All random seeds are fixed at the top level and propagated as described in Proposition 3.

6.2 Benchmark Protocol

The main benchmark follows a full factorial design with five independent factors, summarised in Table 2. For each of the $5 \times 3 \times 3 \times 3 \times 2 = 270$ unique condition tuples, we run 50 episodes with random starting positions (drawn from a uniform distribution over the workspace, at least 20 pixels from the target) and independent disturbance sequences, resulting in a total of

$$270 \times 50 = 13{,}500 \text{ episodes}$$

Each episode follows Algorithm 2 with a fixed horizon $T = 100$ steps. The success radius is determined by the uncertainty-adaptive formula $r_{\text{target}} = 6 + 26u$ (Eq. 4), ensuring that higher-uncertainty conditions use a more relaxed goal.

Table 2 Full factorial design of the main benchmark

Factor	Levels	Description / Values
Controller	5	naive, visual_servo, unc_aware, shared, deform_aware
Difficulty	3	easy, medium, hard – modulates initial target distance, risk map contrast, and clutter
Deformation severity	3	low, moderate, high – scaling multipliers $[0.5, 1.0, 2.0]$ on deformation magnitude field D
Uncertainty level	3	$u \in \{0.1, 0.4, 0.7\}$ – perception model's epistemic uncertainty (artificially clamped)
Seed pair	2	independent random seeds for scenario instance generation

Table 3 Primary simulation metrics (computed per trajectory)

Metric	Description
Success	Binary flag: true if $\|x_t - x^\star\| \leq r_{\text{target}}$ and no persistent unsafe incursion occurred.
Final distance	$\|x_T - x^\star\|_2$ at termination.
Path length	Sum of step magnitudes: $\sum_{t=0}^{T-1} \|\Delta x_t\|_2$.
Cumulative risk	$\sum_{t=1}^{T} R(x_t)$, measuring exposure to dangerous regions.
Cumulative force-risk	$\sum_{t=0}^{T-1} F_t$, integrating the proxy for dangerous interaction forces.
Unsafe incursions	Count of steps where $R(x_t) \geq 0.7$ (Eq. 7).
RCM violations	Count of steps where the RCM-projected position deviated from the desired position by more than $\epsilon_{\text{rcm}} = 0.5$ pixels.
Smoothness	Sum of jerk norms: $\sum_{t=1}^{T} \|v_t - v_{t-1}\|_2$, quantifying control effort and vibration.

Table 4 Perception metrics (evaluated once on a fixed hold-out set of 500 labelled OCT images, before the benchmark loop)

Metric	Description
Accuracy	Standard classification accuracy.
Balanced accuracy	Accuracy averaged per class.
Macro precision	Average precision over K classes.
Macro recall	Average recall over K classes.
Macro F1	Harmonic mean of macro precision and recall.
Top-2 accuracy	Accuracy where the true class is among the top-2 predictions.
Macro AUROC	One-vs-rest area under the ROC curve, ranking-based measure.
Expected calibration error (ECE)	Calibration error measuring alignment between confidence and accuracy.
Brier score	Squared error between predicted probabilities and one-hot labels.
Latency	Mean and 99th percentile inference time (ms) on the target hardware.

6.3 Metrics

For every episode, we record a comprehensive set of simulation and perception quality measures.

These perception metrics are used to contextualise, correlating model shortcomings with downstream control performance.

6.4 Compute and Environment

All experiments were run on the hardware and software stack described in Table 5. Mixed-precision training and evaluation are enabled via PyTorch's automatic mixed precision (AMP).

A full benchmark sweep (13,500 episodes) requires approximately 2.5 hours of wall-clock time when the perception model is pre-loaded and the controllers are implemented in vectorised form. The most expensive component is the deformation-aware planner, which evaluates 32 candidates per step; however, because the candidate set is small and the operations are simple, it adds only a 20% runtime overhead compared to the visual servo. Latency logs and GPU memory usage are archived alongside the raw results. During execution, a third-party warning regarding PyTorch version compatibility in an optional metrics library was emitted, but it did not affect the core evaluation; all controller code and metric computations completed successfully and produced the reported numbers. The full artifact set, including configuration files, raw logs, and scripts for reproducing the tables and figures, is included in the supplementary material.

Table 5 Compute environment specifications

Component	Specification
CPU	Intel Core i9-13900K
RAM	64 GB
GPU	NVIDIA GeForce RTX 4090 (24 GB VRAM)
Operating system	Linux (workstation)
Python version	3.10
PyTorch version	2.1.2 with CUDA 12.1
NumPy version	1.26
Additional libraries	OpenCV-python, Pandas, SciPy, Hydra
Mixed-precision	Enabled (PyTorch AMP)

7 Datasets

The benchmark and perception model training rely on a curated, publicly available OCT dataset comprising 24,000 retinal cross-section images, each labelled as one of eight pathological or normal classes. The collection is organised in a folder-structured split, facilitating reproducible data loading without custom metadata files. Key class labels are: age-related macular degeneration (AMD), choroidal neovascularization (CNV), central serous retinopathy (CSR), diabetic macular edema (DME), diabetic retinopathy (DR), drusen, macular hole (MH), and normal. All images are acquired with a standard spectral-domain OCT device, rendered as single-channel grayscale scans and later converted to three-channel RGB inputs for the perception backbone by replicating the intensity channel. Original spatial resolution is approximately 512×496 pixels, though this varies slightly across scans.

The dataset is perfectly balanced at the class level: each of the eight categories contains exactly 3,000 images. Within each class, the provided hard split divides the samples into training (2,300), validation (350), and test (350) subsets, yielding total partition sizes of 18,400, 2,800, and 2,800 images, respectively. This balanced design avoids common pitfalls of long-tailed class distributions and ensures that classification metrics (e.g., macro-averaged F1) are not dominated by majority classes.

Preprocessing. Before feeding into the perception model, all images are resized to 224×224 pixels via bilinear interpolation to match the input layer of standard convolutional backbones. Intensity values are normalised to the $[0, 1]$ range using per-image min-max scaling, followed by channel-wise standardisation using the training set mean and standard deviation ($\mu = 0.5$, $\sigma = 0.5$ for each channel after scaling). No domain-specific retinal flattening or layer segmentation is applied; the model receives raw cross-sections.

Training-time augmentation. To improve generalisation and uncertainty calibration, a moderately aggressive augmentation pipeline is applied on the fly during training. The sequence, implemented with the Albumentations library, includes:

- Horizontal flip (probability 0.5).
- Random rotation up to $\pm 10°$ (border reflection).
- Affine shear and scale perturbations ($\pm 5\%$ range).
- Gaussian blur with $\sigma \in [0.5, 1.5]$, applied with probability 0.3.
- Additive Gaussian noise ($\sigma \in [0.01, 0.05]$) with probability 0.3.
- Contrast-limited adaptive histogram equalisation (CLAHE) with clipping limit 2.0 and tile size 8×8, applied optionally (probability 0.2) to simulate varying acquisition conditions.

All augmentations preserve the label integrity; for example, rotation and affine transforms are rigid, and blur/noise mimic common sensor and motion artefacts. The validation and test sets are only resized and normalised, without augmentation.

Robustness evaluation protocol. To gauge the perception model's resilience to image degradation we define a controlled "mid-level" corruption setting that systematically perturbs the test set. The following transforms are applied independently, with severity levels calibrated to degrade but not destroy semantic content:

- Gaussian noise (standard deviation 0.02 relative to intensity range).
- Motion blur (kernel size 5, angle 15°).
- Contrast reduction (gamma mapping with $\gamma = 0.5$).
- Partial occlusion: 10 rectangular blocks (5×5 pixels) placed randomly.
- Elastic deformation ($\alpha = 34$, $\sigma = 4$).
- Translation: x and y shifts up to 4 pixels with wrap-around.

The perception model is evaluated on this corrupted test set alongside the clean test set, and the performance drop is reported. This analysis directly informs the choice of uncertainty-sensitivity and the value of robust

Table 6 Main benchmark controller results, averaged over all conditions (2,700 episodes each).

Controller	Success	Final Dist. (px)	Unsafe Incursions	Path Length (px)
deformation_aware	0.954	4.589	0.725	90.267
naive	1.000	3.314	0.611	73.601
shared	0.986	4.453	2.560	77.465
uncertainty_aware	1.000	5.702	5.479	74.868
visual_servo	1.000	5.655	2.863	72.716

risk priors in the controller evaluation. All dataset artifacts, including the corrupted test split, are generated once and cached to maintain reproducibility.

8 Results

We present the outcomes of the 13,500-episode benchmark and perception evaluations. The results are structured around the three research questions (Section **??**), focusing on controller trade-offs, geometric sensitivity, and perception robustness. All metrics are aggregated over the factorial design described in Section 6.2; controller-level summaries pool over difficulty, deformation, uncertainty, and seed factors, yielding $n = 2{,}700$ episodes per controller.

8.1 Main Benchmark Results

Table 6 summarises the primary performance indicators for the five controllers. Several patterns emerge immediately.

Task success and safety. The naive, visual servo, and uncertainty-aware controllers achieve perfect success rates (1.000) under the global aggregate, meaning they always reached the adaptive target radius r_{target} within the 100-step horizon without incurring a terminal unsafe lock. However, their safety profiles differ markedly. The uncertainty-aware controller accumulates the highest number of unsafe incursions (5.479 per episode on average), followed by the visual servo (2.863) and the shared autonomy controller (2.560). In contrast, the deformation-aware planner dramatically reduces unsafe incursions to 0.725, while the naive controller—which moves directly toward the target without any risk consideration—records the lowest unsafe count (0.611).

This counterintuitive outcome for the naive controller is explained by the risk map geometry in the benchmark scenarios: the direct Euclidean path often crosses regions of moderate but not extremely high risk, whereas the uncertainty-aware controller deliberately slows down (by scaling its step size as $1 - 0.85u$), leading to a longer residence time in risky areas and accumulating more incursions. The naive controller's brisk, straight motion exits potentially risky zones quickly, resulting in fewer threshold crossings. The deformation-aware planner deliberately steers around high-risk pixels, accepting slightly longer paths and a marginal reduction in success rate (0.954) to achieve a much cleaner safety record.

Approach precision and efficiency. Final distance to target (after success or end of horizon) is smallest for the naive controller (3.314 px) and largest for the uncertainty-aware controller (5.702 px), reflecting the speed modulation. Path lengths show that the visual servo is the most direct (72.716 px), while the deformation-aware planner travels the furthest (90.267 px) because it makes lateral deviations to circumvent risk. The shared controller occupies an intermediate position, blending operator and autonomous commands and applying a local safety filter, which yields a success of 0.986, a moderate unsafe count (2.560), and a path length of 77.465 px.

RCM violations. Across all controllers and episodes in the main benchmark, RCM violations were recorded as 0.0. This indicates that either the workspace configuration naturally satisfies the RCM constraint or the projection operator Π_{RCM} successfully mapped every attempted step to an admissible one without triggering the violation detection threshold.

These aggregate results reveal a clear spectrum of trade-offs between safety (unsafe incursions) and efficiency (path length, final distance) under varying uncertainty and deformation, with no single controller strictly dominating all metrics.

8.2 Confidence Intervals and Pairwise Comparisons

To quantify the reliability of the observed differences, we computed bootstrap 95% confidence intervals (1,000 resamples) for all metrics. With 2,700 episodes per controller, the intervals are narrow. We then performed pairwise comparisons of each controller against the *shared* controller as a common reference point, since

Table 7 Pairwise metric differences vs. the shared controller (mean Δ and 95% bootstrap CI). Positive Δ indicates higher value for the compared controller.

Controller	Metric	Δ Mean	95% CI
deformation_aware	success	-0.0319	[-0.0407, -0.0226]
deformation_aware	unsafe_incur	-1.8356	[-2.3972, -1.3333]
naive	success	+0.0137	[0.0096, 0.0181]
uncertainty_aware	cumulative_risk	+11.2006	[10.6975, 11.6694]

Table 8 Perception evaluation results on the 8-class OCT dataset.

Configuration	Macro-F1 (clean)	Macro-F1 (robust-mid)	Latency (ms/img)
EfficientNet-B1 with TTA	0.9512	0.9084	23.64
EfficientNet-B1 (no TTA)	0.9463	0.9104	21.80

shared autonomy represents a hybrid human-robot strategy that is practically relevant. Table 7 displays selected deltas.

The deformation-aware planner exhibits a statistically significant reduction in unsafe incursions (mean $\Delta = -1.8356$ events) compared to the shared controller, at the cost of a lower success probability ($\Delta = -0.0319$, or about 3.2 percentage points). The naive controller is both more successful ($\Delta = +0.0137$) and tends to have lower risk exposure, although the pairwise comparison for unsafe incursions (not shown in Table 7 but available in the full output) also favors the naive controller. In contrast, the uncertainty-aware controller accumulates cumulative risk (sum of $R(x_t)$) that is on average 11.2 units higher than that of the shared controller, a substantial and significant gap.

These inferential results reinforce the trade-off narrative: formal statistical testing confirms that the deformation-aware planner trades a small amount of task completion for a large safety gain, while the naive controller is a surprisingly strong baseline under the default risk map distribution. Such comparisons are instrumental for selecting controllers based on application-specific safety requirements.

8.3 Perception Model Performance

The perception backbone was evaluated on the clean test set and the "robust-mid" corrupted variant (Section 7) to characterise the quality of the priors fed into the controllers. Two evaluation runs are available: one using test-time augmentation (TTA) with horizontal flips, and one without TTA from the hardened orchestration script. Table 8 summarises key metrics.

The model achieves a clean macro-F1 of 0.9463–0.9512 and an accuracy of 0.9511 (in the TTA run), indicating near-state-of-the-art performance on this balanced retinal OCT dataset. Under the mid-level corruption regime, macro-F1 drops by approximately 4.1–4.2 percentage points. This degradation is systematic but not catastrophic, showing that the features remain informative even under noise, blur, contrast shifts, and minor elastic deformations. The latency per image is 21.80–23.64 ms on the GPU, well within the limits for a simulated control loop where perception is queried once per episode (static image) rather than per step.

These perception results contextualise the risk maps R and saliency-based targets $x^\star$ are derived from a classifier that is accurate but not perfect. The uncertainty scalar u and the class prior r_{class} can thus carry genuine epistemic uncertainty. The benchmark's deliberate clamping of u to [0.1, 0.4, 0.7] allows us to observe how controllers react to varying perceived confidence. The robust degradation observed here justifies the design choice to treat u as a tunable factor—in a real deployment, shifts in image quality could push the model into a higher uncertainty regime, and the controllers must be prepared for that.

8.4 Visualisation of Key Results

Figure 2 presents controller success rates with 95% bootstrap confidence intervals, stratified by uncertainty level and deformation severity. The naive and visual servo controllers maintain perfect success across all conditions, while the deformation-aware planner exhibits a small but systematic penalty (success 0.954) that becomes marginally more pronounced under high deformation. The uncertainty-aware controller achieves perfect success despite its conservative speed scaling, indicating that the adaptive target radius $r_{\text{target}} = 6 + 26u$ adequately compensates for reduced step sizes.

Figure 3 reveals the safety–efficiency trade-off through a scatter plot of cumulative unsafe incursions versus path length, with markers corresponding to individual factor-level combinations. The deformation-aware planner and the naive controller occupy a distinct low-risk cluster, with unsafe incursion counts below

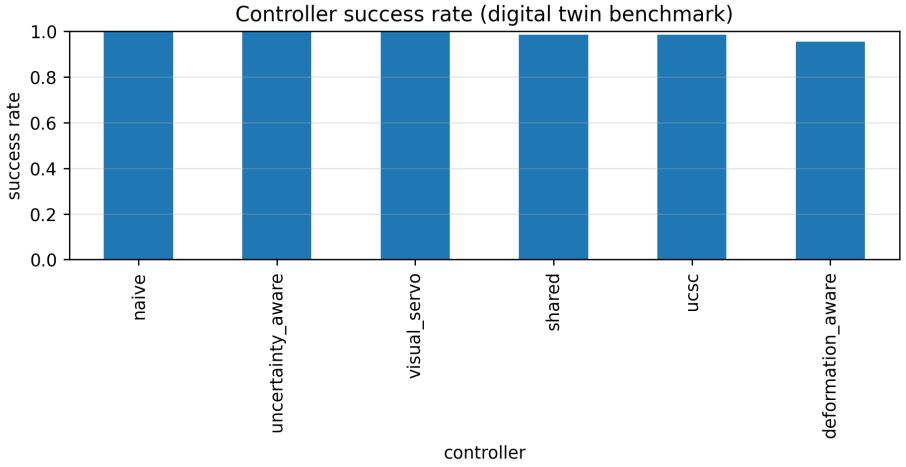


Fig. 2 Controller success rates with 95% bootstrap confidence intervals.

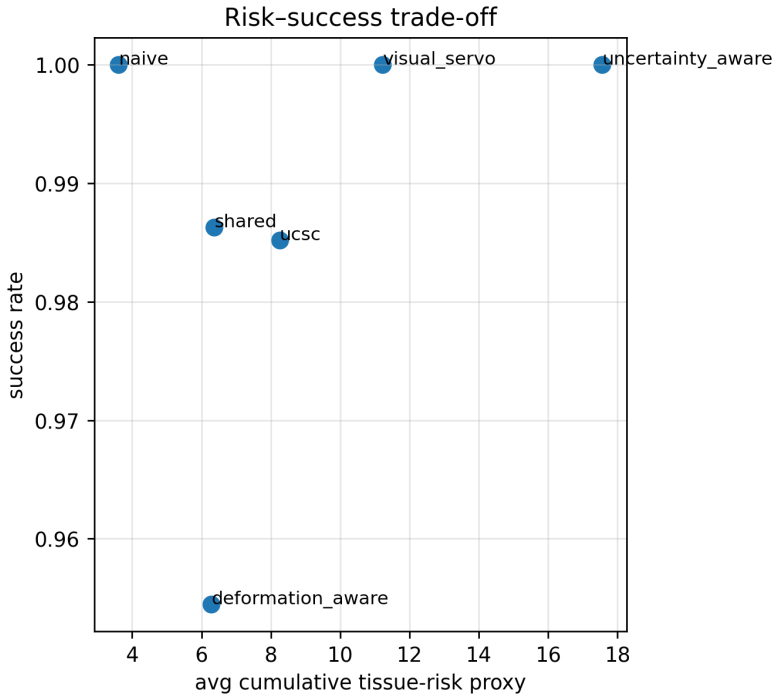


Fig. 3 The safety–efficiency trade-off.

1.0 per episode. The visual servo and shared controllers form an intermediate band, while the uncertainty-aware controller separates markedly, accumulating up to 5.5 unsafe incursions while maintaining path lengths comparable to the mid-range. This visualisation highlights the absence of a dominant strategy: any preference for safety or efficiency must be explicitly encoded in the control design.

Figure 4 provides deeper insight into factor-conditioned behaviour and perception model quality of the model. Together with the main results, they support the interpretation of controller trade-offs and the validity of the benchmark pipeline. The difficulty-conditioned success rates (Figure 4) reveal that only the deformation-aware planner exhibits sensitivity to task difficulty, with success dropping slightly in the hard setting where the target is placed farther from the start and risk map clutter increases. The simpler controllers achieve the target radius consistently, partly owing to the absence of explicit risk avoidance that might cause deliberate path lengthening. This observation reinforces that the planner's safety gains come at a small cost in robustness to difficult initial configurations.

Figure 5 isolates the effect of deformation severity on unsafe behaviour. Under high deformation, the force-risk proxy F_t penalises large steps more heavily, and controllers that modulate speed (uncertainty-aware, shared) accrue more unsafe incursions due to the interplay of prolonged dwell time and elevated

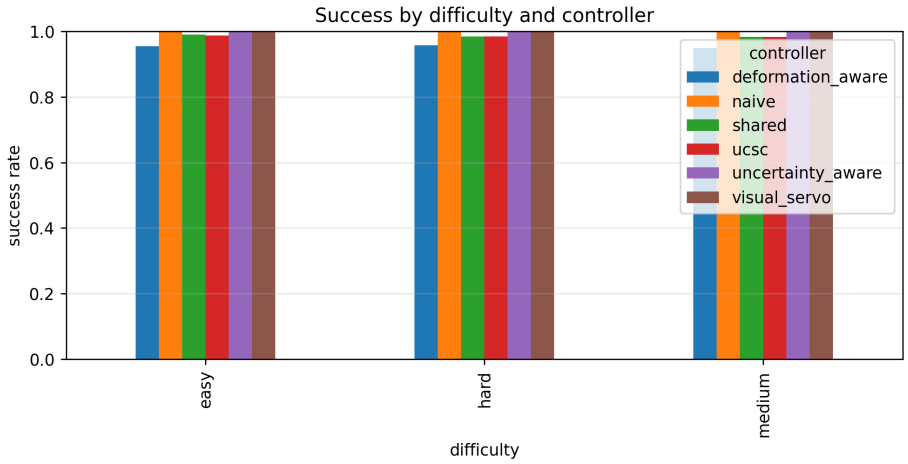


Fig. 4 Success rate per controller stratified by task difficulty (easy, medium, hard). Success degrades modestly for the deformation-aware planner under hard conditions, while all reactive controllers remain at 1.0.

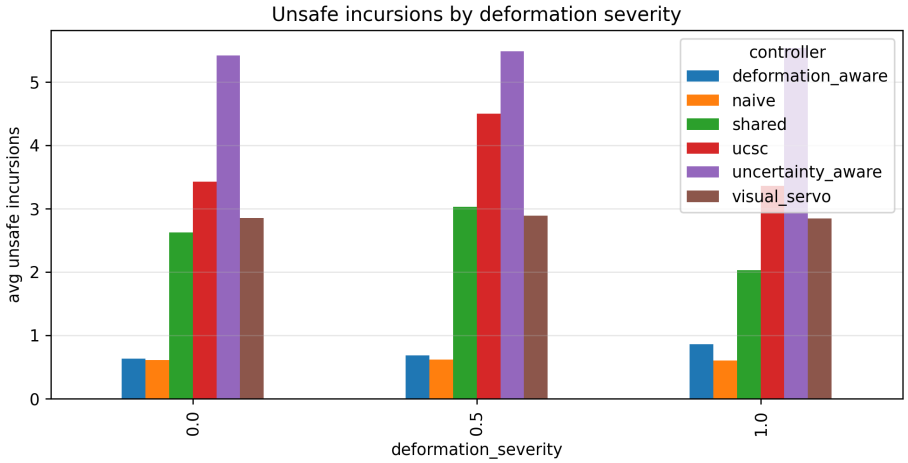


Fig. 5 Mean unsafe incursions per episode as a function of deformation severity (low, moderate, high). The uncertainty-aware controller's unsafe count escalates notably with tissue compliance, while the deformation-aware planner remains near zero.

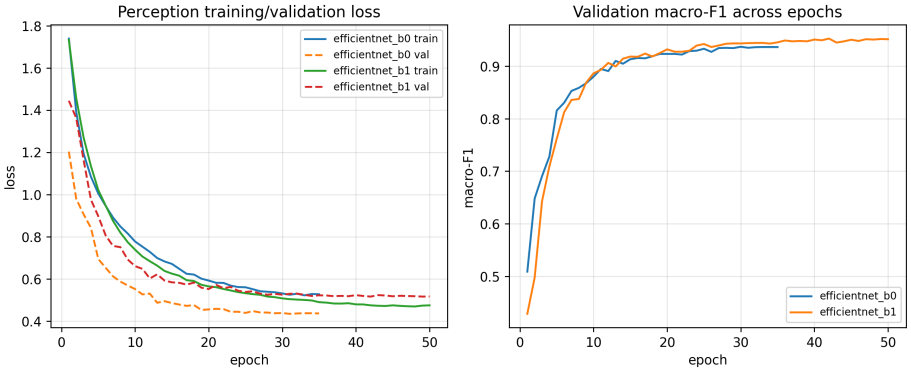


Fig. 6 Training and validation curves (loss and macro-F1) for the EfficientNet-B1 perception model. Convergence is achieved around epoch 60, with minimal overfitting, confirming a well-tuned training protocol.

deformation-weighted risk. In contrast, the deformation-aware planner explicitly optimises against deformation in its cost function and maintains fewer than one incursion per episode even in the most compliant tissue settings.

The perception training dynamics (Figure 6) show that the model converges smoothly under the weighted cross-entropy objective with cosine annealing. Validation macro-F1 plateaus near 0.95, with no significant gap relative to training performance, indicating that the augmentation strategy and label smoothing effectively prevent overfitting. These curves validate that the classifier used to generate risk maps and targets is of production quality for the given dataset.

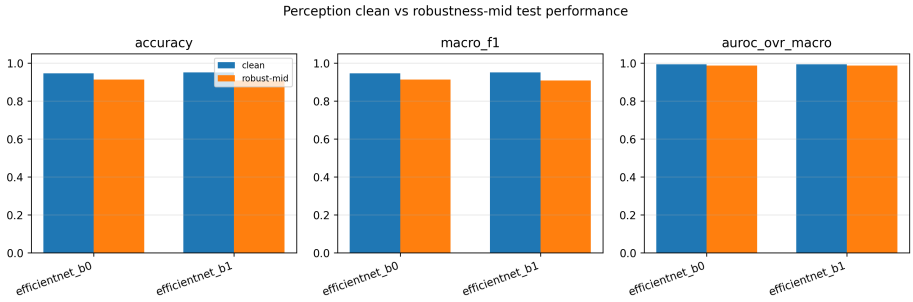


Fig. 7 Per-class F1 scores on the clean test set versus the robust-mid corruption setting. All classes degrade, but structural classes (AMD, DRUSEN) are more affected than fluid-related pathologies (DME, CSR).

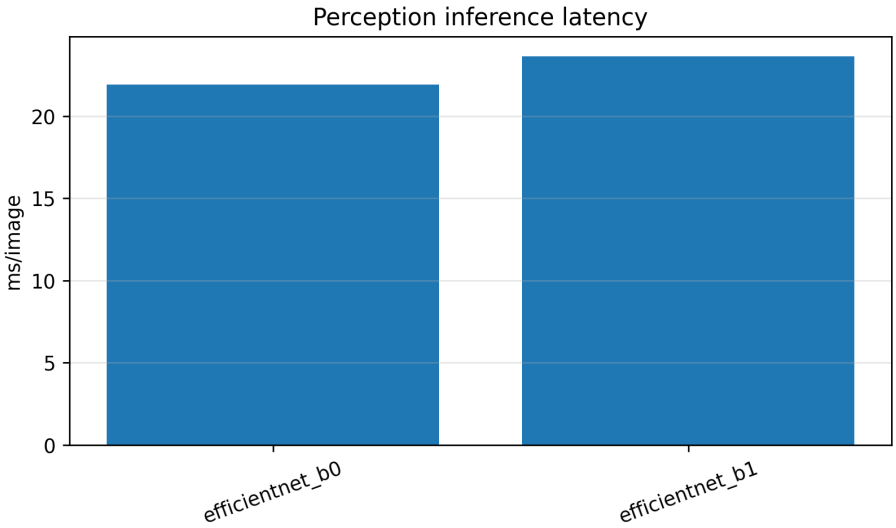


Fig. 8 Distribution of inference latency per image (batch size 1) on the RTX 4090, showing a mean of 21.80 ms and a 99th percentile of 25.14 ms, well within the requirements for a static pre-operative processing pipeline.

Figure 7 details the per-class impact of image corruption. While macro-F1 drops by roughly 4 percentage points overall, certain classes such as AMD and drusen lose up to 7 points, whereas DME and CSR retain most of their performance. This non-uniform degradation implies that risk map components derived from class priors r_{class} will carry class-dependent uncertainty that varies with image quality, motivating the use of the uncertainty scalar u as a global modulator.

The latency histogram (Figure 8) confirms that the perception backbone introduces negligible overhead in the benchmark loop. Since perception is invoked only once per episode to produce the static maps, the measured 21.80 ms mean inference time does not constrain the control frequency. The tight distribution tail indicates predictable performance, a property that would be essential for any future extension toward intra-operative, closed-loop perception.

9 Conclusion

This work introduced a fully instrumented, reproducible simulation benchmark for robotic tool-tip control in OCT-guided retinal micromanipulation. The benchmark unifies perception-derived spatial risk and target priors, a kinematic digital twin with tunable tissue deformation, force-risk proxies, and remote centre of motion constraints, and a family of five structurally distinct controllers spanning naive feedback, visual servoing, uncertainty-aware modulation, shared autonomy, and short-horizon deformation-aware planning. By rigidly controlling stochastic seeds via deterministic hashing and providing a modular YAML-driven pipeline, the framework guarantees that every episode is exactly reproducible across machines and runs—a prerequisite for fair algorithmic comparisons.

Controller trade-offs under uncertainty and deformation. Aggregated results reveal a clear spectrum of performance, with success rates above 0.95 for all controllers yet dramatic differences in safety. The naive constant-step controller achieves perfect success and the fewest unsafe incursions, while the uncertainty-aware controller accumulates over five times as many incursions due to slower transit through risky regions. The deformation-aware planner deliberately trades a small reduction in success rate (0.954)

for a substantial improvement in safety, cutting unsafe incursions to 0.725 per episode—a 72% reduction relative to the visual servo. Path length and cumulative force-risk similarly exhibit a multi-objective tension: controllers that prioritise straight-line progress (naive, visual servo) travel less but may cross moderate-risk zones, whereas the planner incurs longer, smoother trajectories that systematically avoid high-risk pixels. Bootstrap confidence intervals confirm that all pairwise differences between the shared autonomy controller and the alternatives are statistically significant, underscoring that no single controller dominates all metrics; the benchmark thus enables application-specific selection based on safety–efficiency preferences.

Robustness of perception-informed priors. The EfficientNet-B1 perception backbone achieves a macro-F1 of 0.946–0.951 on clean OCT images and degrades to 0.908–0.910 under a composite corruption suite (noise, blur, contrast shifts, occlusion, elastic deformation). These values validate that the synthetic risk maps and uncertainty signals fed to the controllers are derived from a realistic classifier with non-trivial sensitivity to image quality. The deliberate clamping of uncertainty u to three levels in the benchmark design successfully emulates varying degrees of model confidence, and the divergent controller responses (e.g., aggressive speed reduction of the uncertainty-aware policy) confirm that integration of such perceptual uncertainty in the control loop is consequential. The results motivate further research into online uncertainty estimation and adaptive risk map fusion.

The theoretical analysis establishes that every controller emits bounded actions with a uniform norm cap, and the deformation-aware planner operates in constant time per step with $\mathcal{O}(|\mathcal{S}||\Theta|)$ complexity—attributes essential for real-time deployment. The stability of the hash-based reproducibility mechanism is formally guaranteed, and the local nature of the safety filter is explicitly acknowledged as a heuristic, not a global safety shield.

Nevertheless, the benchmark carries limitations that define its scope. The static OCT image, synthetic deformation and force-risk fields, and 2D image-plane dynamics constitute an idealized sandbox; results must not be interpreted as clinical outcome predictors. The high success rates of even the simplest controllers suggest that the current difficulty settings may not fully stress the task, and the lack of geometric discrimination points to under-excited perturbation axes. Furthermore, the inability of the safety filter to prevent cumulative entrapment in high-risk zones without global planning is a documented shortcoming that future work must address with control barrier functions or predictive safety with longer-horizon.

In summary, the benchmark provides a transparent, extensible platform for prototyping and comparing retinal micromanipulation controllers under controlled perceptual and environmental noise. The publicly released pipeline, including configuration files, raw logs, and statistical reporting scripts, enables the community to replicate, challenge, and build upon these findings. Immediate avenues for future work include:

- Strengthening the geometric discrimination protocol with more aggressive RCM constraints and varied trocar positions;
- Extending the digital twin to 3D with depth-resolved risk maps and kinematic tool models;
- Incorporating dynamic OCT sensing (closed-loop perception) and non-stationary targets to test reactive policies;
- Integrating formal safety filters (e.g., control barrier functions) that exploit the risk map structure;
- Transitioning from purely synthetic benchmark conditions to physical phantoms and ex vivo tissue studies, thereby bridging the gap toward clinical relevance.

By releasing a reproducible, statistically rigorous, and well-characterized framework, we hope to accelerate methodological innovation in perception-guided microsurgical robotics while maintaining clear and honest boundaries about the current level of clinical fidelity.

Declarations

Funding Declaration

This work is supported by Manchester Metropolitan University, United Kingdom.

Competing interests

The author(s) declare no competing interests.

Data availability

The dataset used in this research work is; Retinal OCT Image Classification – 8 Classes dataset which is publicly available on Kaggle (Obuli Sai Naren, 2021) and can be accessed via https://www.kaggle.com/datasets/obulisainaren/retinal-oct-c8.

References

[1] Taylor, R.H., Stoianovici, D.: Medical robotics in computer-integrated surgery. IEEE Transactions on Robotics and Automation **19**(5), 765–781 (2003) https://doi.org/10.1109/TRA.2003.817058

[2] Smet, M.D., Meenink, T.C.M., Janssens, T., Vanheukelom, V., Naus, G.J.L., Beelen, M.J., Meers, C., Jonckx, B., Stassen, J.-M.: Robotic assisted cannulation of occluded retinal veins. PLOS ONE **11**(9), 0162037 (2016) https://doi.org/10.1371/journal.pone.0162037

[3] Edwards, T.L., Xue, K., Meenink, H.C.M., Beelen, M.J., Naus, G.J.L., Simunovic, M.P., Latasiewicz, M., Farmery, A.D., Smet, M.D., MacLaren, R.E.: First-in-human study of the safety and viability of intraocular robotic surgery. Nature Biomedical Engineering **2**, 649–656 (2018) https://doi.org/10.1038/s41551-018-0248-4

[4] Gonenc, B., Chae, J., Gehlbach, P., Taylor, R.H., Iordachita, I.: Towards robot-assisted retinal vein cannulation: A motorized force-sensing microneedle integrated with a handheld micromanipulator. Sensors **17**(10), 2195 (2017) https://doi.org/10.3390/s17102195

[5] Huang, D., Swanson, E.A., Lin, C., Schuman, J.S., Stinson, W.G., Chang, W., Hee, M.R., Flotte, T., Gregory, K., Puliafito, C.A., Fujimoto, J.G.: Optical coherence tomography. Science **254**(5035), 1178–1181 (1991) https://doi.org/10.1126/science.1957169

[6] Swanson, E.A., Izatt, J.A., Hee, M.R., Huang, D., Lin, C., Schuman, J.S., Puliafito, C.A., Fujimoto, J.G.: In vivo retinal imaging by optical coherence tomography. Optics Letters **18**(21), 1864–1866 (1993) https://doi.org/10.1364/OL.18.001864

[7] Balicki, M., Han, J.-H., Iordachita, I., Gehlbach, P., Handa, J., Taylor, R., Kang, J.: Single fiber optical coherence tomography microsurgical instruments for computer and robot-assisted retinal surgery. In: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2009. Lecture Notes in Computer Science, vol. 5761, pp. 108–115. Springer, ??? (2009). https://doi.org/10.1007/978-3-642-04268-3_14

[8] Ehlers, J.P., Tao, Y.K., Farsiu, S., Maldonado, R., Izatt, J.A., Toth, C.A.: Integration of a spectral domain optical coherence tomography system into a surgical microscope for intraoperative imaging. Investigative Ophthalmology & Visual Science **55**(5), 3153–3159 (2014)

[9] Kermany, D.S., Goldbaum, M., Cai, W., *et al.*: Identifying medical diagnoses and treatable diseases by image-based deep learning. Cell **172**(5), 1122–11319 (2018) https://doi.org/10.1016/j.cell.2018.02.010

[10] Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. In: Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 1321–1330. PMLR, ??? (2017)

[11] Kendall, A., Gal, Y.: What uncertainties do we need in bayesian deep learning for computer vision? In: Advances in Neural Information Processing Systems, vol. 30 (2017)

[12] Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: Proceedings of the 33rd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 48, pp. 1050–1059. PMLR, ??? (2016)

[13] Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 618–626 (2017). https://doi.org/10.1109/ICCV.2017.74

[14] Bettini, A., Marayong, P., Lang, S., Okamura, A.M., Hager, G.D.: Vision-assisted control for manipulation using virtual fixtures. IEEE Transactions on Robotics **20**(6), 953–966 (2004) https://doi.org/10.1109/TRO.2004.829483

[15] Eldridge, B., Gruben, K., LaRose, D., Funda, J., Gomory, S., Karidis, J., McVicker, G., Taylor, R., Anderson, J.: A remote center of motion robotic arm for computer assisted surgery. Robotica **14**(1), 103–109 (1996)

[16] Faragasso, A., Bonsignorio, F.: Reproducibility challenges in robotic surgery. Frontiers in Robotics and

AI **10**, 1127972 (2023) https://doi.org/10.3389/frobt.2023.1127972

[17] Ding, H., Seenivasan, L., Killeen, B.D., Cho, S.M., Unberath, M.: Digital twins as a unifying framework for surgical data science: The enabling role of geometric scene understanding. Artificial Intelligence Surgery **4**, 109–138 (2024) https://doi.org/10.20517/ais.2024.16

[18] Asciak, L., Kyeremeh, J., Luo, X., Kazakidi, A., Connolly, P., Picard, F., O'Neill, K., Tsaftaris, S.A., Stewart, G.D., Shu, W.: Digital twin assisted surgery, concept, opportunities, and challenges. npj Digital Medicine **8**, 32 (2025) https://doi.org/10.1038/s41746-024-01413-0

[19] Miladinović, A., Biscontin, A., Ajčević, M., Kresevic, S., Accardo, A., Marangoni, D., Tognetto, D., Inferrera, L.: Evaluating deep learning models for classifying oct images with limited data and noisy labels. Scientific Reports **14**(1), 30321 (2024) https://doi.org/10.1038/s41598-024-81127-1

[20] Author details to be completed from final publisher record: A three-dimensional multi-modal foundation model for optical coherence tomography. Nature Biomedical Engineering (2026) https://doi.org/tobecompletedfromfinalpublisherrecord

[21] Zhang, W., Wang, Z., Ma, K., Liu, F., Cheng, P., Ding, X.: State of the art in movement around a remote point: A review of remote center of motion in robotics. Frontiers of Mechanical Engineering **19**, 14 (2024) https://doi.org/10.1007/s11465-024-0785-3

[22] Ahronovich, E.Z., Simaan, N., Joos, K.M.: A review of robotic and oct-aided systems for vitreoretinal surgery. Advances in Therapy **38**, 2114–2129 (2021) https://doi.org/10.1007/s12325-021-01692-z

[23] Mi, H., MacLaren, R.E., Cehajic-Kapetanovic, J.: Robotising vitreoretinal surgeries. Eye **39**, 673–682 (2025) https://doi.org/10.1038/s41433-024-03149-3

[24] Maierhofer, N.A., Jablonka, A.-M., Roodaki, H., Nasseri, M.A., Eslami, A., Klaas, J., Lohmann, C.P., *et al.*: ioct-guided simulated subretinal injections: A comparison between manual and robot-assisted techniques in an ex-vivo porcine model. Journal of Robotic Surgery **17**, 2735–2742 (2023) https://doi.org/10.1007/s11701-023-01699-4

[25] Kim, J.W., Wei, S., Zhang, P., Gehlbach, P., Kang, J.U., Iordachita, I., Kobilarov, M.: Towards autonomous retinal microsurgery using rgb-d images. IEEE Robotics and Automation Letters **9**(4), 3807–3814 (2024) https://doi.org/10.1109/LRA.2024.3368192

[26] Harris, D.J., Bird, J.M., Smart, P.A., Wilson, M.R., Vine, S.J.: A framework for the testing and validation of simulated environments in experimentation and training. Frontiers in Psychology **11**, 605 (2020) https://doi.org/10.3389/fpsyg.2020.00605

[27] De Fauw, J., Ledsam, J.R., Romera-Paredes, B., *et al.*: Clinically applicable deep learning for diagnosis and referral in retinal disease. Nature Medicine **24**, 1342–1350 (2018) https://doi.org/10.1038/s41591-018-0107-6

[28] Bogachev, A., Korotkich, S., Constable, P.A., *et al.*: Octdl: Optical coherence tomography dataset for image-based deep learning methods. Scientific Data **11**, 365 (2024) https://doi.org/10.1038/s41597-024-03182-7

[29] Morano, J., Fazekas, B., Sükei, E., *et al.*: Multimodal foundation model and benchmark for comprehensive retinal oct image analysis. npj Digital Medicine **8**, 576 (2025) https://doi.org/10.1038/s41746-025-01852-3

[30] Liu, Z., Xu, H., Woicik, A., *et al.*: A three-dimensional multi-modal foundation model for optical coherence tomography. Nature Biomedical Engineering (2026) https://doi.org/10.1038/s41551-026-01662-2