Reinforcement Learning for Automated Spectrometer Calibration

Motivation X-ray spectroscopy allows scientists to analyze the electronic and chemical properties of materials by observing how they emit X-rays under excitation. This process relies on precise alignment of spectrometer components, particularly the reflective crystals that direct fluorescence signals onto a detector. Manual alignment of these components is time-consuming and depends on expert intuition, which makes it difficult to scale and optimize beamline usage. Our project aims to automate this calibration process using reinforcement learning (RL), reducing beamtime overhead and improving reproducibility. We focus on developing and validating an RL-based pipeline for aligning the von Hamos spectrometer at the Stanford National Accelerator Laboratory (SLAC), moving toward real-world deployment.

Method We model alignment as a sequential decision-making problem, where the agent receives feedback from detector images and learns how to adjust the crystal to improve alignment. Our approach combines a physics-based baseline controller with a residual reinforcement learning policy. The baseline controller uses local linearizations of the system to perform coarse corrections. The residual policy, trained using Soft Actor–Critic (SAC), learns to refine these corrections, particularly in cases involving nonlinear misalignments such as pitch and defocus. This hybrid design leverages prior knowledge while still allowing the agent to adapt to more complex scenarios. It also helps ensure sample efficiency, which is essential in high-cost lab environments where data collection opportunities are limited.

Implementation We updated an existing simulator to better reflect the real spectrometer setup. This included adjusting motor step sizes, introducing hardware constraints, and developing a reward function based on signal centering and sharpness. We benchmarked Q-Learning, SARSA, and Dyna-Q in this environment to evaluate their performance and validate the simulation. On the real system, we implemented and tested the baseline controller during beamtime. It was effective at correcting yaw, where the mapping between motor action and signal shift is approximately linear. However, it could not correct vertical misalignment or improve focus, highlighting the need for a learned residual policy.

Results In simulation, Dyna-Q consistently outperformed the other tabular methods, showing faster convergence and higher rewards. The baseline controller, when tested on the real spectrometer, successfully corrected yaw but failed to address more complex alignment errors. The residual SAC policy, trained entirely in simulation, improved performance significantly: the final alignment error decreased from 0.073 mm (baseline only) to 0.039 mm with the residual policy. These improvements were achieved using relatively few training steps, demonstrating strong sample efficiency.

Discussion Our experiments show that a hybrid approach can effectively automate alignment in realistic settings. While the baseline controller handles simple corrections well, it lacks the flexibility to manage nonlinear effects. The residual RL policy compensates for these limitations and does so without requiring full online training. Although we were not able to deploy the complete residual controller on the real spectrometer due to beamtime constraints, testing the baseline controller in practice helped validate our design and confirm the need for learning-based corrections. The results in simulation strongly support future deployment.

Conclusion We developed and evaluated a hybrid control pipeline for automated spectrometer alignment that combines domain knowledge with reinforcement learning. Real-world tests of the baseline controller confirmed that simple corrections are possible with linear methods, but also exposed their limits. Our residual policy, trained in simulation, improved alignment accuracy while maintaining sample efficiency. These findings suggest that learning-based control can offer practical advantages in experimental environments like SLAC and lay the groundwork for future deployment of fully automated calibration systems.

Reinforcement Learning for Automated Spectrometer Calibration

Amine Lamouchi
ICME
Stanford University
aminelam@stanford.edu

Martina Del Gaudio
ICME
Stanford University
mdgaudio@stanford.edu

Abstract

Accurate spectrometer alignment is important for high-quality X-ray spectroscopy, but current manual methods are slow and rely on expert knowledge, which is costly given limited beamtime. We propose a hybrid control method that automates alignment by combining a simple physics-based controller with a residual reinforcement learning policy. The baseline handles basic linear corrections, while the residual policy, trained with Soft Actor–Critic (SAC), improves alignment in harder cases like pitch and focus. We tested our method both in a realistic simulator and on the von Hamos spectrometer at SLAC. Real-world use showed the limits of the baseline, while the RL policy, trained only in simulation, greatly improved accuracy and sample efficiency. This work shows how combining domain knowledge with RL can lead to practical and efficient automation in scientific settings, using ideas from Physics-Informed Deep Learning to add useful structure to the learning process.

1 Introduction

X-ray spectroscopy is a technique used to study the electronic and chemical structure of materials by analyzing the X-rays emitted under high-energy excitation. The typical experimental setup involves three main components: a source that excites the sample, a reflective element such as a crystal that disperses the emitted X-rays, and a detector that captures the resulting signal. One widely used geometry for dispersive spectroscopy is the von Hamos configuration, in which a cylindrically bent crystal reflects the X-rays onto a position-sensitive detector. This setup enables high-resolution energy measurements and is used in various instruments, including the MFX beamline at SLAC.

To obtain usable data, the crystal must be aligned such that the reflected signal is both focused and centered within a specific region of interest (ROI) on the detector. A substantial portion of the allocated beamtime is often spent performing this alignment, especially when multiple crystals are used, as each one introduces additional degrees of freedom to tune. Currently, alignment is performed manually by beamline engineers, relying on iterative adjustments and expert intuition. This process is not only time-consuming but also reduces the effective time available for scientific data collection. Our work contributes to ongoing efforts at SLAC to automate this alignment procedure in order to make more efficient use of beamtime.

To automate alignment, we formulate the task as a sequential decision-making problem. Each crystal has two to three controllable degrees of freedom (DoFs), including yaw (rotation), pitch (tilt), and translation (displacement along the beam path). These parameters influence the position and focus of the X-ray signal on the detector. The objective is to adjust the DoFs such that the signal is centered within the ROI and its sharpness is maximized. Reinforcement learning offers a framework in which an agent can iteratively refine its actions based on feedback from detector images, enabling the development of a control policy that improves alignment over time.

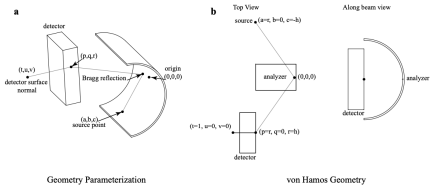


Figure 1: von Hamos spectrometer.

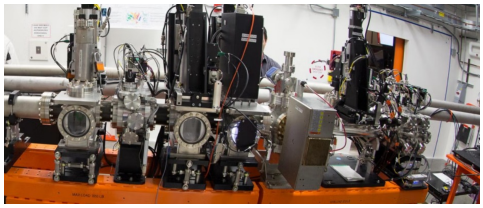


Figure 2: The MFX instrument at SLAC.

We implement this as a Markov Decision Process (MDP) and simulate the environment using the X-ray tracing package `xrt`, which models interactions between the source, crystal, and detector. The simulator is wrapped in a Gym-compatible interface to facilitate integration with RL algorithms. To reduce the sim-to-real gap, we introduced constraints reflecting hardware limitations, such as non-uniform motor step sizes and bounded action spaces. We evaluated three tabular RL algorithms: Q-Learning, SARSA, and Dyna-Q. All methods converged rapidly in simulation, demonstrating that the task is learnable in simplified settings. These results motivated initial efforts toward deployment on the real instrument.

Direct online training on the physical system, however, is infeasible due to safety considerations and the high cost of beamtime. Instead, we adopt a sample-efficient residual learning strategy. We first constructed a baseline controller based on finite difference approximations and a linear system response model. This controller was tested during beamtime on May 23, and successfully corrected yaw misalignments, which exhibit approximately linear behavior. However, it failed to correct vertical misalignment and defocus effects, which are more nonlinear in nature (e.g., due to Bragg angles and the geometry of the Rowland circle). These results motivate the use of a residual policy trained in simulation to handle such nonlinearities.

We then evaluated the complete residual reinforcement learning pipeline in simulation. We compared three configurations: the baseline controller alone, a standard SAC policy trained from scratch, and the hybrid residual SAC approach. Our experiments showed that the residual policy outperformed both the baseline and vanilla SAC in terms of alignment accuracy and sample efficiency. These findings support the use of residual reinforcement learning as a viable method for automating spectrometer alignment under real-world constraints.

2 Related Work

Our starting point is the work of Fuller et al. [1]. They first derive an analytic mapping from the nine spectrometer configuration parameters (source, cylindrical crystal, and detector positions) to the observed fluorescence manifold. Then, they perform calibration by solving for the inverse mapping (i.e., determining the configuration parameters that yield a desired fluorescence pattern).

The forward mapping is mathematically expressed as:

$$F : \{p_1, \ldots, p_9\} \to \{(x_i, y_i)\}_{i=1}^{N} \tag{1}$$

where $p_1, \ldots, p_9$ represent the nine parameters of the spectrometer, $\{(x_i, y_i)\}_{i=1}^{N}$ is the fluorescence manifold and F is the analytically constructed mapping. Calibration is achieved by inverting this mapping to determine the optimal spectrometer settings for a desired fluorescence signal.

$$F^{-1} : \{(x_i, y_i)\}_{i=1}^{N} \to \{p_1, \ldots, p_9\} \tag{2}$$

A significant limitation of this approach is that optimizing for the inverse mapping is highly sensitive to initial conditions and requires parallel multi-start optimization.

More broadly, control problems in robotics and instrumentation have increasingly adopted reinforcement learning as a data-driven alternative, enabling agents to learn adaptive policies directly from observations. Residual reinforcement learning, in particular, has been successful in scenarios where

prior knowledge or approximate models are available, allowing a learned policy to refine the output of a fixed baseline controller [2]. Inspired by these developments, we frame spectrometer alignment as a sequential decision-making problem and explore RL-based solutions that can generalize across configurations and reduce beamtime overhead.

3 Method

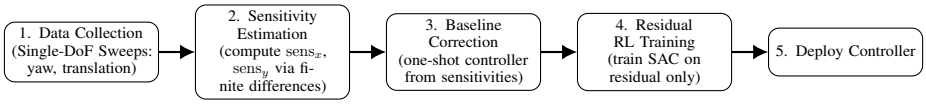


Figure 3: Method Overview.

We formulate spectrometer calibration as an MDP. The goal is to iteratively adjust the crystal's motor configuration to center the reflected X-ray signal on the detector and focus it. The formulation used in our project is as follows:

- **State:** $s_t = (x_t,\ y_t,\ \psi_t,\ T_t)$, where (x_t, y_t) is the current signal position on the detector, ψ_t is the analyzer's yaw angle (in degrees), and T_t is the analyzer's translation along the beam axis (in millimeters).
- **Action:** $a_t = (\Delta\psi_t,\ \Delta T_t) \in [-1, 1]^2$, representing normalized control inputs. These are scaled to physical units via:

  $$\Delta\psi_t = A_\psi \cdot \Delta\psi_t, \quad \Delta T_t = A_T \cdot \Delta T_t,$$

  where $A_\psi = 0.1^\circ$ and $A_T = 2\,\text{mm}$.
- **Transition:** The next state is determined by applying the action to the motor configuration, then querying the simulator (or the actual instrument) to compute the resulting signal position:

  $$\psi_{t+1} = \psi_t + \Delta\psi_t, \quad T_{t+1} = T_t + \Delta T_t,$$
  $$(x_{t+1}, y_{t+1}) = f(\psi_{t+1},\ T_{t+1}),$$

  where $f(\cdot)$ denotes the X-ray tracing simulation function.
- **Reward:** The reward encourages alignment accuracy, penalizes excessive motor movement, and provides a bonus for achieving tight alignment:

  $$r_t = -\|(x_t, y_t) - (0, 0)\|_2 - \lambda\|a_t\|_1 + R_{\text{bonus}} \cdot \mathbf{1}\{\|(x_t, y_t)\|_2 < \varepsilon\},$$

  where $\lambda = 0.01$, $\varepsilon = 0.5\,\text{mm}$, and $R_{\text{bonus}} = 10$.

We implemented this simulation using the `xrt` X-ray tracing package [3]. The simulation is constructed to respect physical constraints, including realistic motor resolutions and bounded action spaces to reduce the sim-to-real gap and ensure that policies trained in simulation remain compatible with the hardware setup at SLAC.

We tested tabular reinforcement learning algorithms such as Q-Learning, SARSA, and Dyna-Q in this environment. These algorithms demonstrated fast convergence and successfully minimized alignment error, validating the environment design and the suitability of the problem formulation. In subsequent stages, we evaluated policy learning strategies that build upon this MDP, including residual reinforcement learning approaches discussed in Section 5.

4 Experimental Setup

We conducted experiments in both simulation and on the physical von Hamos spectrometer at the MFX beamline at SLAC. The simulation environment enabled fast iteration and algorithm development, while real-world testing provided a way to validate key components under realistic conditions.

4.1 Simulation Environment

The simulator was built using the `xrt` X-ray tracing package [3]. We modified the original setup, which modeled a movable source and detector, to instead control the crystal's yaw, pitch, and translation, matching the degrees of freedom actuated during real alignment. This ensured that control commands issued in simulation were consistent with thos e available on the actual instrument.

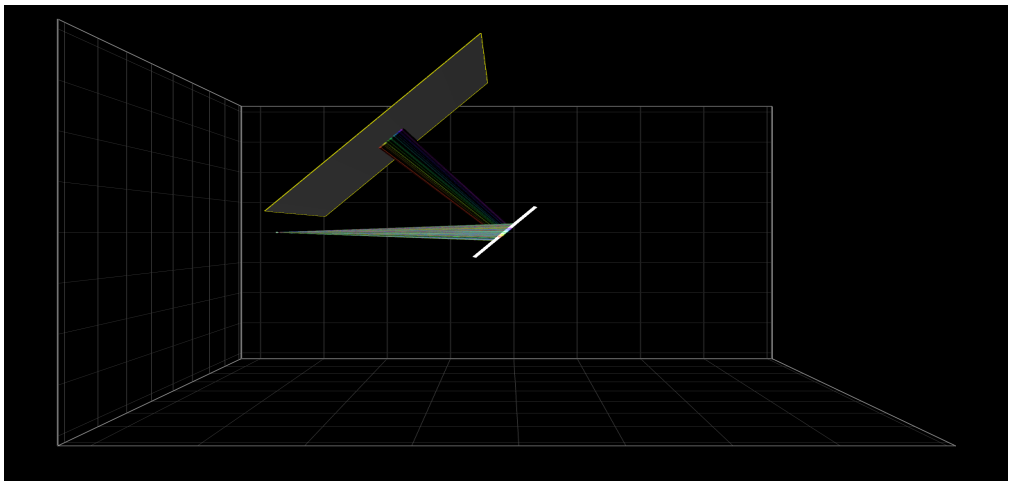


Figure 4: 3D model of the von Hamos spectrometer in simulation.

The simulator was wrapped in a Gym-compatible API, which allowed us to evaluate reinforcement learning algorithms directly. We tested Q-Learning, SARSA, and Dyna-Q on this environment. All three methods converged (as shown in n Figure 5b) and produced policies capable of accurate alignment, validating both the environment and problem formulation.

4.2 Real-World Deployment

We performed three beamtime sessions at the MFX beamline to assess the practicality of our approach. The first two sessions focused on data collection through single-axis motor sweeps across yaw, pitch, and translation. At each configuration, we recorded the resulting detector images. This data allowed us to estimate local Jacobians by finite differences, quantifying how the signal position varied with changes in motor input. We used this information to construct a baseline controller that performs coarse alignment corrections based on an initial misalignment.

During the third session on May 23, we deployed the baseline controller on the actual spectrometer. Given a target position $(x^*, y^*) = (0, 0)$ and measured signal location (x, y), the controller applied proportional corrections calculated as

$$\Delta\psi = \frac{x^* - x}{\text{sens}_x}, \quad \Delta T = \frac{y^* - y}{\text{sens}_y}.$$

To evaluate the effectiveness of these corrections, we implemented a real-time reward computation pipeline. Detector images were first preprocessed using Otsu thresholding to segment the X-ray signal from background noise. We then fit a bounding box around the segmented region to extract its center and area, which were used to compute alignment and focus metrics. These values were mapped to a reward using the same function as in simulation.

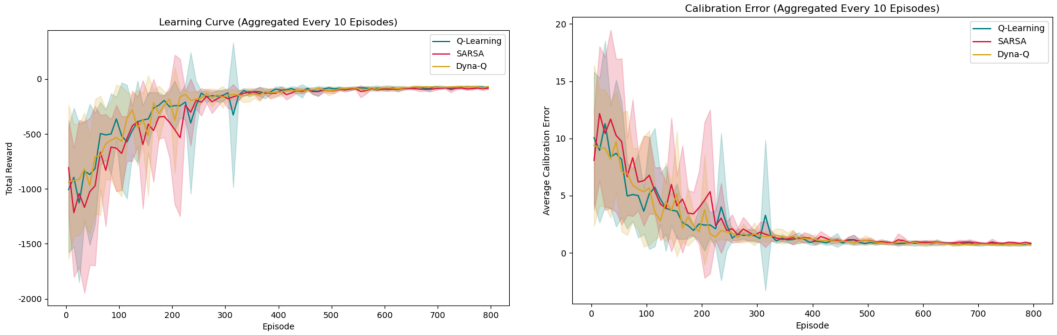
The baseline controller was able to correct yaw misalignments effectively, where the mapping between motor commands and signal movement is approximately linear. However, it failed to correct vertical misalignment and focus, which are governed by more complex physical relationships, such as those arising from Bragg angle sensitivity and the Rowland circle geometry. These limitations highlight the need for a learned residual policy capable of handling such nonlinearities. Although time and safety constraints prevented us from deploying the learned policy on the real system, the baseline tests confirmed that our simulation assumptions were reasonable and that hybrid learning-based methods are a promising direction.

5 Results

We evaluated the proposed hybrid control pipeline in both simulation and on the real spectrometer at SLAC. Our results highlight two key findings: (1) tabular RL methods can solve simplified versions of the alignment task efficiently, and (2) a residual reinforcement learning policy significantly improves alignment accuracy compared to both a physics-based controller and standalone SAC.

5.1 Simulation Results

Tabular Methods. As a first step, we benchmarked Q-Learning, SARSA, and Dyna-Q in simulation. All three methods converged to effective policies within a few hundred episodes. Dyna-Q, which uses a model of the environment, achieved the best sample efficiency and the lowest average alignment error, as shown in Figure 5b. This confirmed that the alignment task is learnable with reinforcement learning and that the simulator provides a reliable testbed.

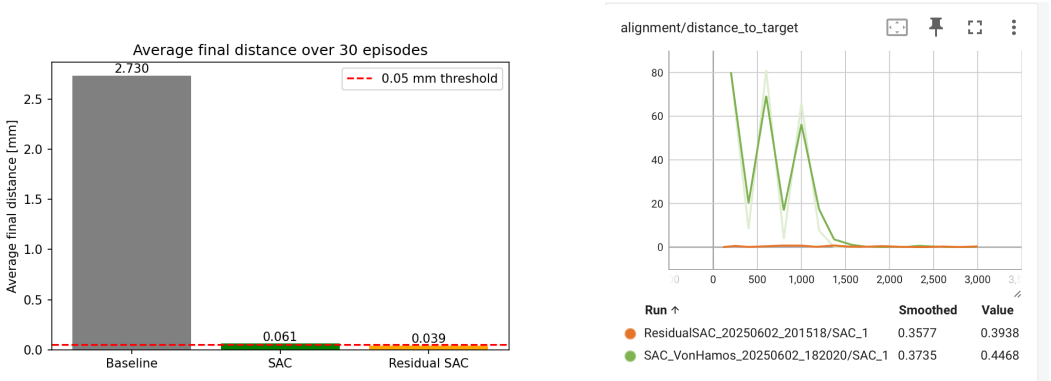


(a) Cumulative reward over training.

(b) Calibration error per episode.

Figure 5: Dyna-Q consistently outperformed SARSA and Q-Learning in terms of convergence speed and final error.

Residual RL. We compared three approaches: the baseline controller alone, a standard SAC agent, and our residual SAC policy. Figure 6a shows that the residual controller achieved the lowest final alignment error (0.039 mm), improving upon the baseline (2.73 mm) and vanilla SAC (0.061 mm). Learning curves in Figure 6b also show that residual SAC converged faster and more smoothly.



(a) Final alignment error.

(b) SAC training curves.

Figure 6: Residual SAC achieves superior accuracy and stability during training.

5.2 Real-World Results

We deployed the baseline controller on the actual spectrometer. It was able to correct yaw errors, thanks to the approximately linear relationship between yaw and horizontal beam displacement.

However, it failed to address vertical misalignment and focus, which require modeling nonlinear effects. Figure 7 illustrates a representative alignment attempt.

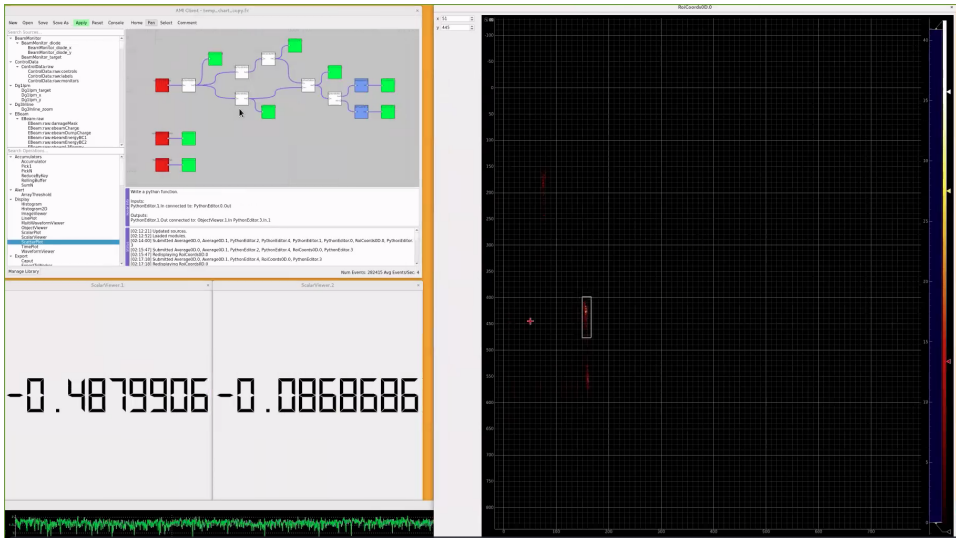


Figure 7: Top left: AMI interface running the baseline controller with real-time detector feedback. Bottom left: motor commands issued to perform the one-shot correction. Right: detector image showing the beam signal (boxed) and the target position (red cross).

These real-world results validate the baseline controller's assumptions and limitations. They further justify using a learned residual policy to handle more complex alignment behaviors, which cannot be captured with linear approximations alone.

6 Discussion

Our results demonstrate that combining domain knowledge with learning-based control leads to significantly improved spectrometer alignment performance, both in simulation and in preliminary real-world testing. The baseline controller, built on finite-difference sensitivities, performed reliably for yaw correction but failed to address vertical misalignments and focus, which depend on nonlinear interactions not captured by linear models. This confirmed our initial hypothesis: a fixed controller alone is insufficient for full alignment and must be augmented with a learned residual.

The residual SAC policy not only improved final alignment accuracy in simulation but also converged faster than a vanilla SAC agent, validating the benefit of injecting prior knowledge into the learning process. Although we were unable to test the full residual controller on the physical spectrometer due to beamtime limitations, the real-world deployment of the baseline controller helped verify the feasibility of our control interface, reward computation pipeline, and calibration methodology.

7 Conclusion

We developed and evaluated a hybrid reinforcement learning pipeline for automating spectrometer calibration at SLAC. Our approach combines a physics-informed baseline controller with a residual policy trained using SAC. In simulation, this residual controller significantly outperformed both the baseline and a standalone SAC agent, achieving sub-millimeter alignment accuracy with strong sample efficiency. Real-world tests of the baseline controller confirmed its utility for simple, linear corrections, while also highlighting its limitations in handling more complex optical misalignments.

These findings support the broader applicability of hybrid learning-based control strategies for scientific instrumentation. Future work will focus on deploying the full residual policy in the lab, improving sim-to-real transfer through domain randomization or digital twins, and extending the framework to multi-crystal systems with coupled degrees of freedom. Our results pave the way

toward fully automated, beamline-ready calibration systems capable of reducing manual overhead and maximizing experimental throughput.

8 Team Contributions

- **Amine Lamouchi**: Implemented the Residual RL pipeline in the simulator; worked on the pipeline for real-detector image processing and reward computation; inspection of previous scans to design the baseline controller.
- **Martina Del Gaudio**: Modified the previous `xrt` simulator to adapt it to the new environment; worked on the pipeline for real-detector image processing and reward computation; inspection of previous scans to design the baseline controller.

References

[1] Fuller et al. "Analytic von Hamos geometry optimization and calibration". [n. d.].

[2] Tobias Johannink, Shikhar Bahl, Ashvin Nair, Jianlan Luo, Avinash Kumar, Matthias Loskyll, Juan Aparicio Ojea, Eugen Solowjow, and Sergey Levine. 2019. Residual reinforcement learning for robot control. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 6023–6029.

[3] Konstantin Klementiev and Roman Chernikov "Powerful scriptable ray tracing package xrt". [n. d.].