SIIM CAIMI26 Conference on Artificial Intelligence in Medical Imaging
OCT 26-27 | University of Pennsylvania, Philadelphia, PA

Beyond Deployment: An In-House System for Continuous Quality Assurance of Deployed Auto-Segmentation Models

Jonathan Cheng, University of Pennsylvania; Nicholas Molina; Kelechi Obike

Problem & Motivation: What clinical or operational problem does your MVP address?
Deploying an AI model is not the last step. In radiation oncology, AI now drafts the organ and tumor contours that determine where radiation is delivered, but once a model is live, the harder question begins: is it still doing what the clinic needs? Deployed models require ongoing observability, and AI introduces new failure modes that traditional quality assurance was never designed to catch. We already QA our linear accelerators on a regular, rigorous schedule. We should be doing the same for our AI models, yet the standards for that don't exist yet. Before we can build them, we need to collect the data and measure how these models are performing over time. That is the real problem. We can't purchase AI tools from a vendor, plug them in, and simply trust that they keep working. Someone has to monitor them, catch the deviations, and make that performance visible to the people responsible for patient safety. Today that monitoring is largely ad-hoc, with no consistent way to measure how closely AI contours agree with the approved clinical contours, where they disagree, or how performance changes across model versions and over time. We built this to bring deployed AI models under the same standard of continuous, measurable quality assurance we already apply everywhere else in the clinic.

What You Built: Describe your MVP, tool, or workflow in plain language
We built two things in-house that work together. The first is dcm-utils, a Python package that reads radiation-oncology medical images and does the underlying geometry and metric calculations. The second is a Dashboard that turns those calculations into a picture of how our AI models are performing. The input is the DICOM data a treatment plan produces: the CT scan, the contours drawn on it, and the delivered dose. A pipeline pulls studies from our Orthanc image archive, separates clinician-approved contours from AI-generated ones using the DICOM approval status, and automatically matches structures across the different naming conventions clinicians and AI models use (for example, a clinician's "L_Lung" against a model's "Lung_L"). For each matched pair, dcm-utils computes a range of comparison metrics, including the Dice coefficient, Hausdorff distance, and surface Dice, along with per-structure shape statistics such as volume. The output is the Dashboard which contains score distributions, per-structure breakdowns, side-by-side comparisons of different AI model versions, and trends over time. It also includes a patient search, which lets us look up a specific patient's contours directly and has been invaluable for diagnosing individual contour problems and catching naming mismatches between clinical and AI structures. Every number shown traces back to a dcm-utils computation, so the Dashboard is a live, query-able view of how our deployed models are doing.

Tech Stack & Development Approach: What tools, frameworks, or methods did you use to build it?
The system has three parts plus the dcm-utils package underneath. The frontend is a React and Vite single-

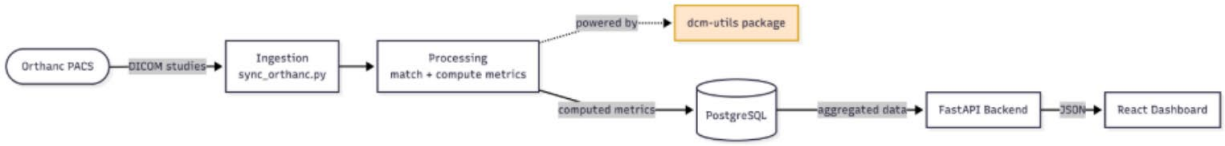
page application with multiple views — clinical, model-comparison, structure-metrics, and patient-search pages — and charts built in Plotly and Recharts. The backend is a FastAPI service backed by PostgreSQL, which serves the aggregated metrics to the frontend. The data pipeline, written in Python, pulls studies from our Orthanc image archive (a free, open-source medical image server), computes the metrics, and stores them. Underneath all of it sits dcm-utils, our in-house Python package, which does the actual DICOM reading and geometry math: turning contours into 3D masks and computing spacing-aware, sub-voxel-accurate overlap and surface metrics. It handles the parts that are easy to get wrong, like anisotropic voxel spacing and coordinate-system conventions, so the pipeline can stay focused on matching, comparing, and storing. One genuinely hard problem was matching structures when clinicians and AI models name the same organ differently. We solved it in stages: a hand-curated alias list, best-overlap matching, exact name matching, and finally fuzzy matching using the Hungarian algorithm (via scipy), which finds the globally optimal one-to-one assignment between two sets of structure names above a similarity threshold. We built the system with Claude Code, which let us move quickly across the entire stack (database migrations, backend endpoints, and frontend charts alike) and iterate on the design as the clinical requirements sharpened.

Validation & Evidence: Have you done any informal testing or validation? If so, what did you find?
The strongest evidence is that the system runs on real clinical data. It ingests actual studies from our hospital Orthanc archive, which contains real CT scans, clinician-approved contours, and AI output. To date, our system has processed 8,097 studies across 7,422 patients. This is production evidence of function on the same data the clinic works with every day. The numbers it reports are trustworthy by construction. Every metric is a well-established segmentation measure (Dice, Hausdorff at the 95th percentile, surface Dice) computed on spacing-aware 3D masks that correctly handle anisotropic voxels and coordinate-system conventions, the details most likely to introduce silent error. Where a metric is undefined, such as an empty mask or a study with no target, the pipeline returns an explicit null rather than a misleading zero, so no fabricated value ever reaches a chart. Automated tests pin these behaviors across the pipeline, from metric storage to the API. Most importantly, the tool has already proven useful in practice. Patient search has let us diagnose individual contour problems and catch naming mismatches between clinical and AI structures that would otherwise go unnoticed, and version-to-version comparison lets us see how a newer model performs against an older one on the same anatomy. Furthermore, we can pinpoint the structures our AI models consistently struggle with and focus our efforts there. The evidence is not a one-time study; it is a system that surfaces real discrepancies on real cases, continuously.

Feedback You're Seeking: What specific feedback, help, or collaboration are you hoping to get from the CAIMI26 community?
We would value your perspective on three questions. First, and most fundamental: is continuous monitoring of deployed auto-segmentation models genuinely useful, and do you see it as widely adoptable across institutions? We believe deployed AI needs ongoing quality assurance the same way a linear accelerator does, but we want to hear whether practitioners see this as a real need or a solution in search of a problem. Second, what agreement thresholds should separate an acceptable AI contour from one that genuinely needs editing, and should those thresholds differ from one structure to another? A Dice of 0.85 may be excellent for a large organ and unacceptable for a small structure near the cord, and we have no consensus on where those lines belong. Third, beyond the metrics already on our roadmap, what features or measurements would take this system to the next level? We have identified the obvious next steps, such as dose-weighted comparison, but we are most interested in the non-obvious ones: the views, signals, or capabilities that a daily user would want but that we have not thought to build.

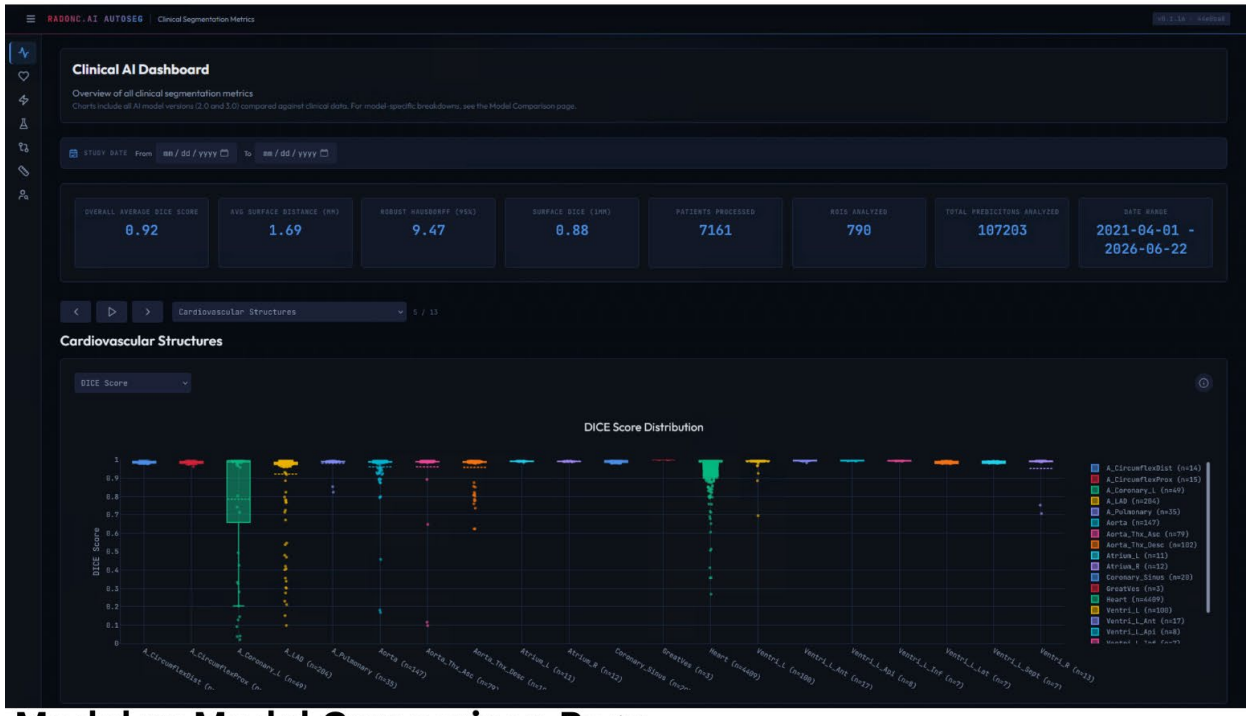
Known Limitations & Honest Failures: What isn't working yet, or what have you already tried that failed? Each of our main limitations comes down to the same gap: the numbers are computed correctly, but reading them still requires clinical context. The first and deepest is our definition of ground truth. We treat any contour with a DICOM approval status of APPROVED as the clinical reference, but physicians sometimes approve an AI contour with little or no editing even when it does not truly qualify as ground truth. When that happens, an AI-versus-clinical comparison is really AI-versus-approved-AI, which inflates agreement and overstates model quality. Such a contour shows near-perfect Dice and near-zero Added Path Length, so the metrics we are adding may themselves become a way to flag these cases. The second is clinical convention. When contouring breast, clinicians often crop several millimeters from the skin, a step the AI model does not know to take. Our system flags this as an AI-versus-clinical disagreement even though it is not a model error. This is the kind of disagreement that convention-aware or distance-weighted comparison is meant to handle, and addressing it is on our roadmap. The third is structure naming. Our multi-stage matcher has made false matches rare, but no matcher can fully compensate for inconsistent naming at the source. A physician focused on one breast may label it simply Breast rather than Breast_L or Breast_R, leaving no way to tell which side it is and risking a match against the wrong side. This is why reliable naming, and the standards behind it, remains an open problem.

Potential for Impact: If this MVP were fully developed and deployed, who would benefit and how? The immediate beneficiary is the whole department, not one role. Our system gives physicists and dosimetrists a shared, current view of auto-segmentation quality by structure and model version, replacing ad-hoc spot checks with a consistent record. It shows physicians where a model is systematically weak, so review attention goes where it is most needed, which is a patient-safety benefit as much as an efficiency one. It gives quality and safety an auditable, continuous account of how a deployed model is performing over time. And it gives administrative leadership objective data for decisions that are otherwise made on vendor claims alone: whether to adopt a model, keep it, or replace it. That last point is not hypothetical. When a newer model version was introduced, our system showed that agreement with approved clinical contours did not improve, and on certain structures it declined. Without continuous monitoring, that regression would likely have gone unnoticed, and the newer version would have been adopted on the assumption that newer means better. This is exactly the failure mode that motivated the project: a deployed model can quietly get worse, and only measurement will tell you. The broadest impact reaches past our own department. Standards for monitoring deployed AI models do not largely exist yet and building them will require real-world performance data collected consistently over time. That is precisely what our system produces, at the scale of thousands of patients. We see it as an early contribution toward the evidence base such standards will need.

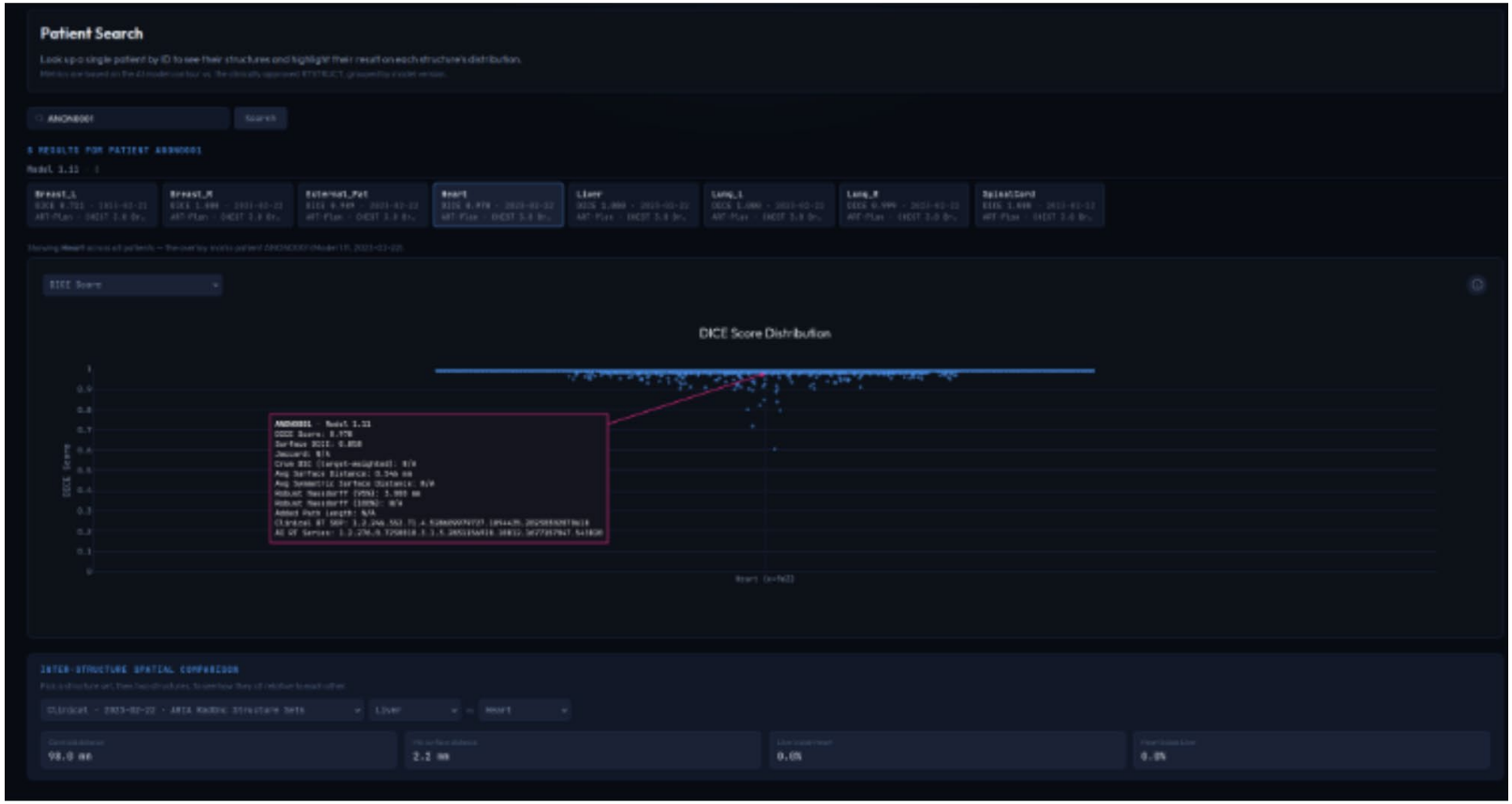

Orthanc PACS
DICOM studies
Ingestion
sync_orthanc.py
Processing
match + compute metrics
powered by
dcm-utils package
computed metrics
PostgreSQL
aggregated data
FastAPI Backend
JSON
React Dashboard


Structure Metrics
Cohort shape metrics for clinical (approved) structures — volume, connected-component count, and internal holes — computed from the contour geometry. Distributions are aggregated across all patients that have a CT and an approved RTSTRUCT.
8,097
7,422
Large (100–1000 cc)
VOLUME DISTRIBUTION — LARGE (100–1000 CC)
Large (100–1000 cc)
Volume (cc)
AVERAGE CONNECTED COMPONENTS
Average Connected Components by Structure (Top 15)
Avg # connected components
PER-STRUCTURE METRICS
STRUCTURES
PATIENTS
(0)BODY ONLY
BODY

All Models vs Clinical Comparison Page



Model vs Model Comparison Page


Patient Search
Look up a single patient by ID to see their structures and highlight their result on each structure's distribution.
ANON0001
Search
6 RESULTS FOR PATIENT ANON0001
Model 1.11
Breast_L
Breast_R
External_Pat
Heart
Liver
Lung_L
Lung_R
SpinalCord
DICE Score
DICE Score Distribution
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
DICE Score
ANON0001 · Model 1.11
DICE Score: 0.978
Surface DICE: 0.850
Jaccard: 0.1%
Crum IOU (target-weighted): 0/0
Avg Surface Distance: 0.546 mm
Avg Symmetric Surface Distance: 0/0
Robust Hausdorff (95%): 1.005 mm
Robust Hausdorff (100%): 0/0
Added Path Length: 0/0
Heart (n=963)
INTER-STRUCTURE SPATIAL COMPARISON
Clinical · 2023-02-22 · ANIA RadOnc Structure Sets
Liver
Heart
98.0 mm
2.2 mm
0.0%
0.0%