

A Two-Validator Web Interface for Structured Geometry Figure Annotation

Sabin-Codruț Badea
Dumitran

SABIN-CODRUT.BADEA@S.UNIBUC.RO and Adrian-Marius
MARIUS.DUMITRAN@UNIBUC.RO

University of Bucharest, Faculty of Mathematics and Computer Science

Abstract

Annotating geometric figures from scanned documents has long been addressed by adapting generic annotation tools, tools not originally designed for such tasks, to use cases where they are suboptimal. An interactive web interface is described that is purpose-built for validating automatically generated geometry figure descriptions, allowing annotators to review and correct conditional declaration language (CDL) descriptions while simultaneously adjusting figure crops and editing source problem text. Submissions pass through two independent annotators in sequence, with each round fully logged. The interface is currently deployed and has been used by 12 annotators to validate 483 problem entries.

Keywords: structured annotation, geometry figures, human-in-the-loop, serverless

Code and Deployment: Implementation and deployment guide available at <https://github.com/badea-codrut-cti/GeoValidator>¹.

1. Introduction

The digitization of olympiad-level geometric corpora is essential for building high-quality datasets that power intelligent tutoring systems, automated exercise generators, and adaptive hint engines for geometry education (Lu et al., 2021, 2024). OCR systems extract problem text with reasonable accuracy, and multimodal models can generate initial figure descriptions, but both stages produce noisy outputs that must be human-validated before they can safely be used in educational applications, because a single mis-labelled diagram or spoiled construction can invalidate an entire worked solution.

Generic annotation platforms (e.g., CVAT, Label Studio, Supervisely) offer robust project management but were not designed for structured mathematical content. Conversely, LaTeX editors such as Overleaf and Mathcha provide syntax validation but no image cropping or diagram annotation capabilities. No existing platform combines geometry-focused image cropping with structured description validation for geometric figures.

This gap matters because geometry reasoning datasets require precise figure-diagram annotations alongside problem text (Lu et al., 2021; Chen et al., 2021). In educational settings, rigorous validation of AI-generated descriptions acts as a lightweight governance layer for training and evaluation corpora, ensuring that downstream tools such as benchmark suites, automated solvers, or tutoring software are grounded in human-verified facts. Quality annotation frameworks like CrowdTruth (Dumitrache et al., 2018) and Headwork (Gross-Amblard et al., 2024) address crowdsourcing quality but do not target mathematical content.

The present work addresses this gap by describing a lightweight, serverless web interface for validating structured geometry figure annotations, deployed on a free-tier edge platform to keep the infrastructure accessible to teachers and small research groups.

1. Repository with the platform source code.

2. System Design

2.1. Interface Layout

The interface uses a three-panel layout: problem statement with MathJax rendering (editable for OCR correction), source image with cropped diagram and Cropper.js re-cropping, and CDL description as plain-text editor (Figure 1).

CDL Validator
Validator 1 · Easy Logout

Batch progress (Easy)
0 / 20 in batch | 1231 remaining in total

Image

Crop Preview

Problem Statement:

Subiectul III. (7 puncte) $\frac{x}{y} = \left(\frac{x}{x+y+z}\right)^2 \Rightarrow \frac{x}{\sqrt{y}} = \frac{x}{x+y+z}$ (2 puncte) $\frac{y}{x} = \left(\frac{y}{x+y+z}\right)^2 \Rightarrow \frac{\sqrt{y}}{x} = \frac{y}{x+y+z}$ (2 puncte) $\frac{z}{y} = \left(\frac{z}{x+y+z}\right)^2 \Rightarrow \frac{\sqrt{z}}{y} = \frac{z}{x+y+z}$ (2 puncte)

$\Rightarrow \frac{\sqrt{x} + \sqrt{y} + \sqrt{z}}{\sqrt{y}} = 1$ Prin urmare

$S = (\sqrt{s_1} + \sqrt{s_2} + \sqrt{s_3})^2$

(1 punct)

CDL Description

```
Triangle(A, B, C)\nPointOnLine(X, Line(B, C))\nPointOnLine(Y, Line(B, C))\nPointOnLine(M, Line(A, B))\nPointOnLine(N, Line(A, C))\nParallel(Line(M, X), Line(A, C))\nParallel(Line(N, Y), Line(A, B))\nPointOnLine(P, Line(M, X))\nPointOnLine(P, Line(N, Y))\nEquals(LengthOf(Line(B, X)), x)\nEquals(LengthOf(Line(X, Y)), z)\nEquals(LengthOf(Line(Y, C)), y)\nEquals(AreaOf(Triangle(M, B, X)), s_1)\nEquals(AreaOf(Triangle(N, Y, C)), s_2)\nEquals(AreaOf(Triangle(P, X, Y)), s_3)\nEquals(AreaOf(Triangle(A, B, C)), S)
```

Figure 1: Screenshot of the three-panel validation interface.

2.2. CDL Notation Development

CDL was originally proposed by Zhang et al. (Zhang et al., 2024) as the conditional declaration language within the FormalGeo framework, consisting of construction, condition, and

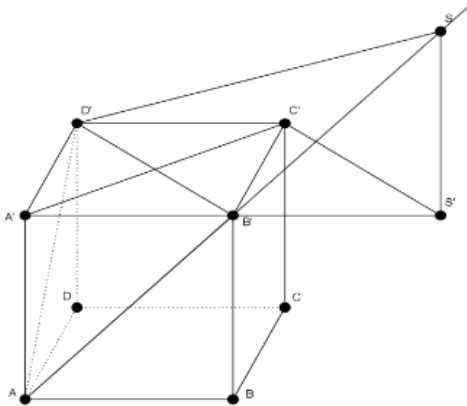
goal statements. In that formal system, CDL is intentionally low-level so that every geometric property is stated atomically. For instance, a parallelogram must be declared through four separate `Parallel` predicates and two pairs of `Equals(LengthOf(...))` statements. While this granularity is essential for automated theorem provers, it becomes prohibitively verbose for multimodal LLMs, which must generate descriptions in a single forward pass and which rely on chain-of-thought reasoning that scales with output length. Higher token counts increase both inference cost and error rate, and the resulting descriptions are harder for human validators to parse.

We therefore deviate from the original FormalGeo CDL by introducing higher-level composite predicates that preserve semantic coverage but reduce statement count. Examples include `Parallelogram(A,B,C,D)` in place of four parallel-line declarations, and `Midpoint(M,Line(A,B))` in place of two collinearity and length-equality statements. These refinements cut average description length by roughly half while remaining expressible in FormalGeo’s underlying predicate logic. Table 1 summarizes the current predicate hierarchy.

Category	Purpose	Example predicates
Shape declarations	2D/3D primitives	<code>Triangle(A,B,C)</code>
Structural relations	Geometric properties	<code>Parallel(Line(A,B),Line(C, D))</code>
Semantic measurements	Numeric constraints	<code>Equals(LengthOf(Line(A,B),6))</code>
Edge cases	Arbitrary/unrepresentable	<code>Equals(..., Latex(\sqrt{3}{62}))</code>

Table 1: CDL predicate categories and representative examples.

The predicate set was refined iteratively over 12 rounds, starting from a minimal subset and expanding based on observed annotation patterns. Of the 483 validated entries, 67 required no modifications to the AI-generated CDL output.



```
% 3D example
Cube(A, B, C, D, A', B', C', D')
Equals(LengthOf(Line(A, B)), 6)
PointOnLine(B', Line(A, S))
PointOnLine(B', Line(A', S'))
Perpendicular(Line(S, S'), Line(A', S'))
Equals(MeasureOf(Angle(D', S, B')), 30)
```

Figure 2: Example of 3D CDL description for a cube geometry problem.

2.3. Two-Pass Validation Workflow

Each entry is reviewed by two annotators sequentially. The first annotator corrects AI-generated CDL descriptions, adjusts figure crops, and edits problem text. Submissions without modifications are recorded as approvals; any edit stores the corrected content in per-annotator fields. The second annotator reviews the same entry with the first’s corrections pre-filled, completing the entry. Every field has first-validator and second-validator columns, preserving original AI output alongside corrections.

3. Limitations and Future Work

Validator agreement was substantial but not perfect. Of the completed entries, both annotators agreed on the final CDL, crop, and problem text in 67 cases (approximately one in seven). Agreement was determined on the basis that no changes were made to the crop, problem statement or CDL description, which was uncommon due to the fact that the first validator could potentially miss a tighter, more accurate crop of the geometric figure or a raw Unicode character in the problem statement that could be rewritten using inline Mathjax notation (i.e $\cap$ instead of $\cap$).

The majority of second-pass corrections to the CDL description were not disagreements over geometric facts, but rather refinements to prevent the description from *spoiling* the problem, which was the most frequent failure mode we observed. For example, when a problem asks the student to *prove* that a quadrilateral is a square, the model often described the figure as `Square(...)` because the drawing *looks* square, even though the predicate itself encodes the conclusion to be proved. Maintaining pedagogically faithful descriptions is critical for educational downstream use: a tutoring system that pre-answers its own exercise, or a benchmark that supplies the theorem under test, undermines learning and invalidates evaluation. The two-pass workflow is deliberately designed to catch such didactic leakage.

Future work includes automated pre-validation steps that detect potential spoilers (e.g., flagging predicates whose consequences match the problem objective), formal inter-annotator agreement statistics on predicate-level disagreements, and extending the pipeline to other national olympiad corpora.

4. Ethics Statement

All geometry problems used in this work were extracted from publicly available Romanian olympiad materials. The annotation platform records which session annotated which images but does not store any personally identifiable information about annotators. All annotators who participated in the validation process were adult volunteers and provided informed consent for the use of their annotation data.

5. Conclusion

We described a web interface for validating AI-generated geometry figure descriptions. The tool combines problem text editing, figure cropping, and CDL review in a two-pass workflow with full edit logging, and it runs on free serverless infrastructure.

References

- Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric P. Xing, and Liang Lin. Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning. In *Findings of ACL*, 2021. URL <https://arxiv.org/abs/2105.14517>.
- Anca Dumitrache, Oana Inel, Lora Aroyo, Benjamin Timmermans, and Chris Welty. Crowdtruth 2.0: Quality metrics for crowdsourcing with disagreement. 2018. URL <https://arxiv.org/abs/1808.06080>.
- David Gross-Amblard, Marion Tommasi, Iandry Rakotoniaina, Constance Thierry, Rituraj Singh, and Leo Jacoboni. Headwork: a data-centric crowdsourcing platform for complex tasks and participants. In *International Conference on Extending Database Technology*, 2024. URL <https://api.semanticscholar.org/CorpusID:266905623>.
- P. Lu, H. Bansal, T. Xia, et al. Mathvista: A multimodal reasoning benchmark for mathematical understanding of visual figures. In *ICLR*, 2024. URL <https://arxiv.org/abs/2310.02255>.
- Pan Lu, Ran Gong, Shibiao Jiang, Liang Qiu, Siyuan Huang, Xiaodan Liang, and Song-Chun Zhu. Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning. In *ACL*, 2021. URL <https://arxiv.org/abs/2105.04165>.
- X. Zhang, N. Zhu, Y. He, et al. Formalgeo: An extensible formalized framework for olympiad geometric problem solving. 2024. URL <https://arxiv.org/abs/2310.18021>.