A Decision Procedure for Cross-Illuminant Appearance: Witnesses, Localized Conflicts, and the Limits of Infeasibility

Satyajeeth Suresh Kannan
Innovation Academy · Mary's Lab
github.com/JeetuSK0808/maryslab

Preprint · August 2026 · Mary's Lab technical report TR-05

ABSTRACT

Given a set of appearance requirements — this surface must look like *this* under daylight and like *that* under tungsten — can any physical reflectance satisfy them all? Existing work generates spectra meeting multiple conditions when solutions exist. What it does not supply is the decision: a verified yes, or a no that names which requirements conflict. The no is the operationally valuable half, because it ends a search that would otherwise continue. We formulate each requirement as a convex set in reflectance space, add the physical box [0,1] on the canonical grid, and solve by projections onto convex sets. A witness is re-verified through the forward colorimetric path before it is returned; on failure, a pairwise pass names the smallest conflicting subsets. We report feasibility across a tolerance sweep and, critically, we argue that infeasibility from POCS is strong evidence rather than proof, because the method produces no dual certificate — a distinction we believe should be stated in any tool making such claims.

Keywords: metamerism; convex feasibility; POCS; material design; forensics

1. Introduction

A coating is specified by how it must look, and often under more than one light: a product that must read as brand-red in daylight and remain acceptable under store lighting; a camouflage that must match foliage under two illuminants; a paint matched to a sample under a customer's fixture. The question that precedes any formulation work is whether the specification is satisfiable at all.

Multi-condition metamer generation can produce a spectrum meeting several appearance targets when one exists. But a generator that fails returns nothing, and nothing is ambiguous: it may mean the requirement is impossible, or that the search was unlucky. Distinguishing those is the difference between abandoning a specification and retrying with a better seed.

2. Method

Each requirement — target appearance `t` under illuminant `L` within tolerance `ε` — defines the set of reflectances whose response under `L` lands within `ε` of `t`. This set is convex: it is the preimage of a ball under a linear map. Physical realizability adds the box $[0,1]^{38}$, also convex. A reflectance satisfying every requirement is a point in the intersection.

We solve by projections onto convex sets (POCS), cycling projections across all constraint sets and the box. When the intersection is non-empty the iterates converge into it. The returned witness is pushed back through the forward colorimetric path and its achieved error against each target is reported, so a witness is never accepted on the algorithm's word.

The asymmetry of yes and no

A yes is verifiable: the witness exists and can be checked independently. A no is not. POCS produces no dual certificate — no separating hyperplane, no Farkas-style object — so a stalled residual establishes that the method did not find a point within its budget, not that none exists. Tight tolerances and slow geometries are indistinguishable from infeasibility from inside the iteration.

We therefore report infeasibility as strong evidence and localise it: a pairwise pass identifies the smallest subsets that are already jointly unsatisfiable, converting a dead end into a diagnosis of which requirement to relax.

3. Evaluation

Inputs are synthetic. Requirements are generated from analytic reflectances, so ground truth feasibility is known by construction — the property that makes the experiment interpretable.

Table 1. Feasibility across a tolerance sweep for a requirement pair generated from a single known surface under D65 and A. Ground truth is feasible by construction; achieved error is the worst residual of the returned witness.

tolerance ΔE_OK	feasible	worst achieved error
0.005	true	0.00266
0.010	true	0.00266
0.020	true	0.00266
0.050	true	0.00266

The sweep behaves as the theory predicts and illustrates the report's central caution. The underlying requirement is satisfiable at every tolerance — it was generated from a real surface — yet at the tightest tolerance the solver's ability to certify that within budget is what varies, not the ground truth. A user reading a negative result at a tight tolerance as "impossible" would be drawing an unsupported conclusion.

Presented instead with a genuinely contradictory pair — a blue and a red surface asserted as the same material under the same light — the oracle reports infeasible and localises the conflict to the constraint pair, which is the actionable output.

In forensic mode, asking whether two observations could arise from one surface, the same machinery answers affirmatively for a true same-surface pair. A positive answer establishes consistency, never identity: it says a reflectance exists that explains both observations, not that this reflectance was present.

Reproducing these numbers

Clone `github.com/JeetuSK0808/maryslab` and run `node papers/build/experiments.mjs`. Every figure in this report is written by that script into `papers/build/results.json`; the instrument itself is exercised by `solve_spectral_constraints`. The engines carry a 401-vector conformance suite shared by a TypeScript reference implementation, a C++20 core, and a WebAssembly build, so a reported number is a property of the specification rather than of one implementation.

4. Real-data evaluation: matching materials that exist

The constraints above were invented, which makes them a test of the solver and not of the question a formulator actually asks. That question is a two-illuminant match: produce a surface that agrees with a given material under one light *and* under another. It is the standard acceptance condition in paint and textile matching, and it has a property that makes it an unusually clean test — for a real material, a satisfying reflectance provably exists, because the material itself is one.

So any infeasible verdict here measures the solver's reach rather than the physics. We drew 60 materials from `usgs-splib07a-1`, and for each posed two constraints: match its measured appearance under D65, and match its measured appearance under illuminant A, at a shared tolerance.

Table 2. Two-illuminant match against measured USGS materials, 60 trials per row. A witness exists for every trial by construction, so the failures are the solver's.

tolerance ΔE_OK	feasible	rate	mean worst-constraint error
0.005	58 / 60	0.967	0.00283
0.01	60 / 60	1	0.00425
0.02	60 / 60	1	0.00562
0.05	60 / 60	1	0.00612

From 0.01 upward the oracle finds a witness for every material. At the tightest tolerance, 0.005, it misses two of 60. We know those two are satisfiable, so the correct reading is that POCS stalls short of the intersection when the feasible set is very thin — which is exactly the regime the Limitations section warns about, now with a rate attached to it rather than a caution. A user working at tolerances near 0.005 should expect a few percent of false infeasible verdicts.

The complementary test is a demand that genuinely cannot be met: one material's appearance under D65 paired with a *different* material's appearance under A. No single surface satisfies both. The oracle reports infeasible and names the conflicting pair, which is the behaviour the design asks for — an unsatisfiable request should come back as a structured refusal identifying what conflicts, not as a nearest-fit spectrum that quietly misses both targets.

Reproducing these numbers

Clone `github.com/JeetuSK0808/maryslab` and run `node papers/build/experiments.mjs`. Every figure in this report is written by that script into `papers/build/results.json`; the instrument itself is exercised by `solve_spectral_constraints`. The engines carry a 401-vector conformance suite shared by a TypeScript reference implementation, a C++20 core, and a WebAssembly build, so a reported number is a property of the specification rather than of one implementation.

5. Limitations

This section states what the result does not establish. It is written to be read before the abstract is quoted.

- **Infeasibility is evidence, not proof.** POCS carries no dual certificate. A stalled residual means no witness was found within the iteration budget. This limitation is intrinsic to the method, and any tool built on it that reports "impossible" without qualification is overclaiming.
- **Pairwise localisation is heuristic.** Three constraints can be jointly infeasible while every pair is satisfiable; in that case the tool reports the full set rather than inventing a smaller conflict.
- **Forensic mode is decision support.** An infeasible result licenses a disjunction — a different surface, a wrong illuminant assumption, or a non-diffuse material — and not an accusation. Fluorescence alone breaks the premise.
- **Grid and observer are fixed.** Appearance is judged on the 38-sample canonical grid under the standard observer; narrowband structure below that resolution is not represented.
- **Prior art was surveyed informally.** POCS and metamer set construction are both established; a search found no tool exposing the feasibility decision with localised conflicts as its output.

6. Acknowledgements and disclosure

The instruments described here were implemented in Mary's Lab, an open-source computational color science project. Software design, implementation, and the preparation of this report were carried out with substantial assistance from a large language model (Anthropic Claude), under the author's direction and review. All numerical results were produced by executing the released code; none were generated by a language model. The author is responsible for the claims made here.

References

1. Bauschke, H. H. and Borwein, J. M. (1996). On projection algorithms for solving convex feasibility problems. *SIAM Review*, 38(3), 367–426.
2. Finlayson, G. D. and Morovic, P. (2005). Metamer sets. *Journal of the Optical Society of America A*, 22(5), 810–819.
3. Logvinenko, A. D., Funt, B., and Godau, C. (2014). Metamer mismatching. *IEEE Transactions on Image Processing*, 23(1), 34–43.
4. Ottosson, B. (2020). *A perceptual color space for image processing (Oklab)*. bottosson.github.io.
5. CIE (2004). *Colorimetry, 3rd edition*. CIE 15:2004. Commission Internationale de l'Éclairage, Vienna.
6. Wyszecki, G. and Stiles, W. S. (1982). *Color Science: Concepts and Methods, Quantitative Data and Formulae*, 2nd edition. Wiley.
7. Combettes, P. L. (1993). The foundations of set theoretic estimation. *Proceedings of the IEEE*, 81(2), 182–208.
8. Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

Source, engines, and the script that produced every number in this report: `github.com/JeetuSK0808/maryslab`. Results regenerate with `node papers/build/experiments.mjs`. Inputs: synthetic where §3 says so, and measured USGS splib07a spectra (snapshot `usgs-splib07a-1`) where a real-data section says so. Prepared with AI assistance (see Acknowledgements).