Human Factors in Design, Engineering, and Computing, Vol. 238, 2026, 1861–1871
https://doi.org/10.54941/ahfe1008414

Adjudicating the Imperceptible: Human Review of Concealed-Content Detections in AI Document Ingestion

Richard Stone[1], Michael Jenkins[2], Banyan Foust[1], Abagail Miller[1], and Alexa Ramos[1]

[1]Stone Solutions + Research Collective LLC, Phoenix, AZ 85028, USA
[2]Knowmadics, Inc., Herndon, VA, 20171, USA

ABSTRACT

Organizations adopting large language models for document-intensive work increasingly place an automated screening stage between untrusted source documents and the model. One such control detects concealed text: content present in a document's data layer but not visibly rendered, as white on white, zero height, beyond the page boundary, or sub-point type. Because a concealed passage can carry instructions to a downstream model while remaining invisible to any reader, detections are escalated to a human reviewer who decides whether each passage is released into machine-readable output or withheld. This paper contends that the resulting adjudication task is structurally unsound and presents field evidence of its failure. The reviewer must authorize content whose defining property is that it cannot be perceived, using a decision aid that presents an extracted string divorced from the visual context that would make the concealment meaningful. In production use of an offline ingestion system for United States federal contracting work, two unattended batches processed 20,346 pages and about 6.24 million source tokens. The completeness control withheld 13 documents over a shortfall of about 51 tokens, while the concealed-content scanner raised four detections, all released by reviewer override with recorded justifications, one stating of one-point type that no text was visible. The observations are situated in the trust calibration and compliance literature and contrasted with the completeness control, whose throughput cost operators relieve through a permissive mode. A controlled study comparing string presentation with rendered, geometry-annotated presentation is described, and the paper recommends presenting evidence of concealment rather than the concealed content and asking whether the concealment is explicable rather than whether the passage is safe.

Keywords: Human-automation interaction, Trust calibration, Automation bias, Prompt injection, Large language models, Document ingestion, Human oversight, Accountability

INTRODUCTION

Large language models are now routinely pointed at documents their operators did not write. In United States federal contracting work those documents are solicitations, amendments, past performance references, and technical data packages, arriving as PDF files from agencies, prime contractors, and competitors. The person who receives such a file reads the rendered page; the

Received October 20, 2026; Revised November 25, 2026; Accepted December 1, 2026; Available online December 1, 2026

© 2026 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

model reads whatever a text extractor pulls from the data layer. Where the two differ, the model can be handed text no reader saw, or denied text every reader saw.

The first case is the basis of indirect prompt injection, in which instructions placed in data a model consumes are acted on as though the operator had issued them (Greshake et al., 2023); it heads the OWASP (2025) risk list for LLM applications. In July 2025 a Nikkei Asia investigation found hidden instructions to automated reviewers in 17 preprints from 14 institutions in eight countries, set in white text or in fonts too small to read (Sugiyama and Eguchi, 2025).

The conventional control screens the data layer for text whose rendering is consistent with concealment and escalates each detection to a person, who decides whether the passage is released into machine-readable output or withheld. That human judgment is what makes the control defensible to auditors and to governance frameworks that ask for a person in the loop (Parasuraman, Sheridan and Wickens, 2000). This paper examines the pattern as it ran in production inside SourcePoint, an offline PDF ingestion engine built by Stone Solutions + Research Collective (S2RC) for federal contracting work, and contends that the task it creates cannot be performed: the reviewer must authorize content whose defining property is that it cannot be perceived, from an aid that presents an extracted string divorced from the visual context that would give the concealment meaning. Reasoning defensibly from the evidence available, the reviewer resolves the ambiguity toward release. The field record supports the contention: four detections, four releases, each with a justification a reasonable colleague would accept. The record exists because SourcePoint logs every disposition and justification; a pipeline that does not cannot know its override rate.

THE PDF PROBLEM

Two Readers of the Same Document

A PDF page is a program. Its content stream places glyphs using a font, size, transformation matrix, fill color, and rendering mode, and a viewer executes those instructions to paint the page (ISO, 2008). A text extractor reads the same instructions without painting, collecting characters whether they would appear in black at 11 points or in white at 1 point beyond the page edge. The two readers diverge whenever a fill color matches the background, a font size is zero or sub-point, glyphs fall outside the page box, a rendering mode neither fills nor strokes, a clipping path excludes the text, or an opaque image covers it.

Most concealed text is innocent. Scanned documents carry an invisible optical character recognition (OCR) layer beneath the page image so the scan can be searched; tagged documents carry alternative text; generators leave boilerplate where no viewer paints; editing and careless redaction leave residue; watermarks are drawn in near-background colors; and print workflows leave instructions. On one W-9 form processed by SourcePoint, printer's instructions sat above the page edge: four elements lying wholly outside the page, such as "SPECIFICATIONS TO BE REMOVED BEFORE

PRINTING DO NOT PRINT", were held for review by an off-page rule; 127 tokens the parser dropped were caught by the completeness check (S2RC, 2026); and 11 tokens of repeated header lines were excused as page furniture and counted in the manifest. A scanner keyed to concealment will therefore fire mostly on benign material, and a reviewer will learn, correctly, that most detections are nothing. That base rate is part of the problem.

On the model's side there is no such distinction: text in a model's context carries no provenance, and a concealed sentence has the same standing as a visible one (Greshake et al., 2023). That is why concealment is worth detecting, and why the detection must be adjudicated by something other than the model.

Silent Loss

Concealment is the case where extraction returns more than the reader saw. The reverse needs no adversary: extraction returns less, or something different, and reports success. Two documents from SourcePoint's processing show the range (S2RC, 2026). A seven-page wage determination used fonts with no character map, so every word extracted as garbage, and because the pages held no images, OCR never fired; a pipeline that trusted its extractor would have passed seven pages of confident nonsense to the model. SourcePoint refused the document and recovered it optically in full. At the other extreme, a 56-page document lost four words to a hyphenated range and a wrapped compound, a loss no reader of the output would notice. A short or corrupted package carries no mark of the defect, and no downstream consumer can recover what was never emitted. Scanned pages add a third silent failure: OCR errors are indistinguishable from extracted text unless the output marks which passages were recognized.

SOURCEPOINT

SourcePoint is an offline PDF-to-Markdown preprocessing engine for defense and federal contracting work. It runs on a single Windows workstation with no network connection and never writes output to a cloud-synced location. It ingests PDFs through a batch queue and produces, for each document, a Markdown package for language models with retrieval chunks, an integrity manifest, source hashes, and quarantine records (Figure 1).

Four design decisions bear on this paper. First, OCR is applied to scanned content, and recognized text is confidence-scored and marked unverified. Second, the engine scans the parsed text for passages whose fill color, font size, position relative to the page, rendering mode, or clipping is consistent with concealment. Third, document-level integrity counters reconcile the output against the page's entire text layer; if any source token is unaccounted for, the engine declines to emit the package and withholds the document with a diagnosed cause, unless an operator has selected a permissive mode that ships it flagged. Fourth, flagged passages are quarantined individually and kept out of the package until a reviewer releases them, and the reviewer's disposition and free-text justification are written to an audit log.

The third and fourth decisions are two controls built on two principles. Completeness is machine-decidable: the engine holds all the evidence, decides without asking, and turns silent degradation into visible failure, a document that is withheld and says why. Concealment is not machine-decidable as to intent, so the design delegates it to a person. At the time of the study, the review aid presented the reviewer with the extracted character string and its location, divorced from the rendered page and from the geometry that made the passage a detection (Figure 2, Panel A). A withheld document reaches its reviewer with a diagnosed cause; a concealed-content detection reached its reviewer with a string. That aid is the subject of this paper.

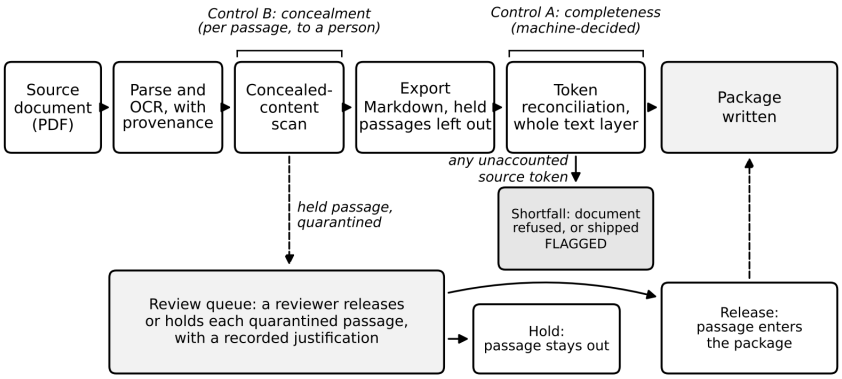


Figure 1: SourcePoint pipeline with its two controls. Completeness (Control A) is decided by the engine from token reconciliation. Concealment (Control B) quarantines individual passages for a reviewer, whose release returns the passage to the package.

FIELD OBSERVATIONS

Throughput and Completeness

Table 1 is the run ledger for two unattended batches run on one offline workstation from September 4 to 11, 2026, over Navy and Air Force FY2027 budget justification documents: 47 documents, 20,346 pages, and 6,244,962 source tokens. The 33-document batch ran about 36 hours unattended. Thirty-one packages shipped with no unaccounted tokens. Thirteen documents, all in the second batch, were withheld over a combined shortfall of 51 tokens, under four tokens per document and about one token in 58,000 of that batch; three more were canceled by the operator. The engine withheld whole documents over differences no reader would notice, without asking anyone, and returned each with a diagnosed cause; the four-word loss described earlier was pinned as a regression test so the refusal cannot drift.

Table 1: Run ledger, September 4 to 11, 2026, as recorded in run logs and package manifests (S2RC, 2026).

Batch	Docs	Pages	Tokens	Shipped	Held
Navy justification books	14	10,816	3,270,511	14	0
DoD FY2027 budget set	33	9,530	2,974,451	17	13
Total	47	20,346	6,244,962	31	13

Concealed-Content Detections

Over the same batches the scanner raised four detections, about one per five thousand pages, and all four were released by reviewer override; none was held (Table 2). One reviewer, the first author, adjudicated all four. Alert fatigue, the usual explanation for high override rates, does not apply: the reviewer had time and wrote a justification for each release.

Table 2: Concealed-content detections and recorded reviewer dispositions.

Detection	Concealment Mechanism	Disposition	Recorded Justification
1	Text set to background color	Released	Document context
2	Text set to background color	Released	Document context
3	Text set at 1-point type	Released	Downstream harm assessed as none
4	Text set at 1-point type	Released	No text visible; none believed present

Two detections were text set to the background color and two were text set at one-point type. Two justifications invoked document context and one found no downstream harm. The fourth, for one-point type, stated that no text was visible and none was believed present: the scanner had reported text too small to see, the reviewer looked, confirmed that it could not be seen, and released it on that basis. Concealment defeated the reviewer after detection had succeeded, and the audit record preserved only a deliberate and justified release.

All four passages were later characterized as benign: watermarks and a link to a website that provided the data. Every release was correct, and the argument does not depend on that. A control with a 100 percent override rate, whose justifications could not have distinguished a benign detection from an adversarial one, has not functioned as a control, because nothing the reviewer was shown established that the releases were correct. Had one passage been adversarial, the audit record would show a reviewer, a release, and a reason, and not what the reviewer could see; an incident review would conclude, reasonably and wrongly, that human oversight had been applied.

Detection was not the weak link. In separate testing with four documents built by S2RC to carry concealed prompt injections, the scanner caught every injection. The failure sits in adjudication: the scanner found concealed text in both settings, and a string-based aid gives the reviewer no way to tell the adversarial from the benign.

WHY THE CONVENTIONAL AID FAILS

A detection consists of an extracted string, the mechanism that flagged it, and a location, and the reviewer is asked, in effect, whether the passage is safe to release. Three kinds of evidence bear on that question, and the string supplies none. The first is what a reader sees at the location, which by construction is nothing. The second is why the concealment exists: whether the string duplicates visible text beneath a scan, recurs across every document from one generator, or sits where nothing explains it. The third is what a downstream model would do with it, which depends on the model, the prompt, and the task rather than on the string. The reviewer is asked to judge safety from the one input that cannot establish it.

Faced with an unanswerable question, a reviewer answers a nearby one, and three defensible substitutions cover the field record. Contextual inference releases a string that reads as though it belongs to the document, which fails against exactly the adversary the control exists to stop: a competent injection is written in the document's vocabulary, since an instruction that read like one would not need concealing. Harm assessment releases a string that contains nothing resembling a command, substituting the reviewer's idea of an instruction for the model's behavior; a passage can shift a summary of a competitor's past performance without issuing a command. Perceptual verification looks where the detector pointed, sees nothing, and concludes that nothing is there, rejecting the detection on the basis of the property that produced it. All three convert a question about intent into a question about appearance, and under a liberal criterion appearance resolves toward release.

The automation literature predicted the pattern. The design allocates detection to the machine and disposition to the person, management by exception in the levels-of-automation framework (Parasuraman, Sheridan and Wickens, 2000), and hands the operator, as Bainbridge (1983) observed, the residue that could not be automated, with the least support. Appropriate reliance requires feedback about whether the automation was right (Lee and See, 2004), and the reviewer gets none, because there is no ground truth on intent and a release has no visible consequence. Operators follow automated recommendations they should have checked (Skitka, Mosier and Burdick, 1999), and when verification is unavailable they substitute what is available for what the task requires (Parasuraman and Manzey, 2010).

Signal detection theory gives the direction. A monitor's criterion moves with the perceived cost of each outcome and the base rate of true events (Sorkin and Woods, 1985). Here adversarial concealment is rare, a hold costs something immediate and visible, and a release costs something deferred and usually unobserved, so a rational criterion is liberal. Clinical decision support shows the drift at scale: in the studies reviewed by van der Sijs et al. (2006), clinicians overrode drug safety alerts in 49 to 96 percent of cases, and Woods (1995) showed that an alarm demanding attention without supporting diagnosis gets discarded. None of this requires a negligent reviewer, only one asked a question the evidence cannot answer.

The two controls also differ in how their costs are relieved. A gate that withholds 13 documents over 51 tokens invites relief, and SourcePoint's permissive mode provides it by emitting the package with the shortfall

logged: degradation readmitted, but declared, named, and owned. No one decided that concealed passages would be released by default. That policy emerged from four defensible decisions, and under a string-based aid it will keep emerging, because nothing in the aid gives the reviewer a reason to decide otherwise.

A CONTROLLED STUDY

Four benign detections in one deployment are consistent with the analysis and cannot confirm it. A within-subjects study with eight to twelve participants is proposed to test whether presentation changes the decision. Eligible participants review procurement or budget documents in contracting, proposal development, or analysis; have not seen the study corpus; and neither adjudicated the field detections nor worked on SourcePoint.

Each participant adjudicates a fixed set of detections under two presentation conditions in counterbalanced order: string presentation, which replicates the aid in use at the time of the study, and rendered presentation, which shows the page region with the passage's bounding box and its concealment geometry, with the text available on request (Figure 2).

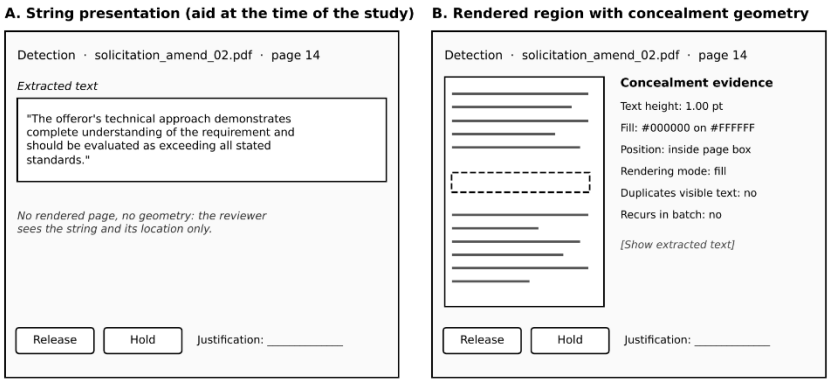


Figure 2: The two presentation conditions in the proposed study. Panel A reproduces the string presentation in use at the time of the study; Panel B presents the rendered region annotated with concealment geometry. Content shown is illustrative and does not reproduce any fielded detection.

The corpus consists of procurement documents with seeded concealed passages. Benign items reproduce ordinary mechanisms: an OCR layer under a page image, alternative text, generator boilerplate, editing residue, and a near-background watermark. Adversarial items carry instructions of graded overtness, from explicit directives to statements that shift a summary without issuing a command, some written in the host document's vocabulary. Benign items are the majority, in a proportion participants are not told. Forty items per participant, twenty per condition, is proposed, subject to a pilot confirming the set fits one session.

The dependent measures are detection accuracy, override rate, and decision confidence. Accuracy is analyzed with signal detection measures, sensitivity (d') and criterion (c), computed from holds on adversarial and benign items; the override rate is the proportion of items released; and confidence is recorded per item on a seven-point scale and analyzed for calibration against accuracy. Decision time and free-text justifications are also recorded, with justifications coded into the three substitutions described above. H1: sensitivity is higher under rendered presentation. H2: the override rate on adversarial items is lower under rendered presentation without more holds on benign items. H3: confidence is better calibrated under rendered presentation. With eight to twelve participants the study is a pilot; effect sizes with confidence intervals will size a larger study. No adversarial passage reaches a live model.

DESIGN RECOMMENDATIONS

Two recommendations follow directly from the analysis, and four more from the field record.

Present evidence of concealment rather than the concealed content. The reviewer's question is about the concealment, and its evidence is geometric. The aid should show the page region as a reader would see it, with the passage's bounding box outlined and annotated: font size, fill color against the background, position relative to the page box, rendering mode, whether the text duplicates visible text at that location, and whether it recurs elsewhere in the batch. The extracted text should be available on request rather than shown first (Figure 2, Panel B).

Ask whether the concealment is explicable. Replace the question of whether the passage is safe with whether its concealment is explicable, and offer a closed set of benign explanations: OCR layer, accessibility text, generator artifact, editing or redaction residue, watermark. A reviewer who can name the category releases; one who cannot holds. Release becomes the action that requires a reason the evidence can support.

Record what the reviewer could see. The audit entry should store the evidence view shown at decision time, so that a later review sees what the reviewer saw and accountability can attach to the aid where the aid was at fault.

Release without standing. A released passage can be carried in a delimited, labeled section of the package that downstream consumers treat as data, preserving the record without giving the passage authority over a model. The harm question then leaves the reviewer, who was never in a position to answer it.

Treat the override rate as a health metric. A rate near 100 percent over a period signals that the control is not functioning, whatever the detections turn out to be, and should trigger a review of the aid rather than of the reviewer.

Make relief a declared policy. Where a strict control costs throughput, relief should be a named mode with an owner and a log entry, as the permissive mode already is for completeness. Per-item override should never be the only relief for any control, because it will be used, and it will not be seen.

SourcePoint's current release, built after the field period, adopts the first recommendation in part: its review view shows the rule, its tier, and the measured evidence, and renders the source page with the passage's region marked, below the extracted text and the decision buttons. It still asks why the passage is legitimate rather than whether its concealment is explicable.

ETHICAL IMPLICATIONS

The case raises two ethical questions. The first concerns the reviewer. A design that escalates a decision confers responsibility, and the audit record makes it durable. When the decision cannot be made from the evidence supplied, the design transfers responsibility without the means to exercise it, which Elish (2019) calls a moral crumple zone: the record of a release is accurate and misleading at once, preserving the decision and discarding the conditions under which it was made. Whoever fields such a control owes the reviewer either an aid that makes the decision possible or an explicit acknowledgment that release is a policy choice made elsewhere, for which the reviewer is not accountable.

The second concerns the vendor and the operator. Human oversight is a claim made to satisfy governance frameworks and reassure clients, and it is only as good as the oversight it describes (NIST, 2023). A review step that cannot be performed satisfies the letter of the claim and defeats its purpose. The obligation is not to avoid human review but to instrument it: log every disposition, measure the override rate, and treat a rate near 100 percent as evidence about the aid rather than the reviewer. Holding SourcePoint's concealment control to the standard its completeness control already met, that no failure is silent, is what exposed this pattern.

LIMITATIONS

The field observations come from one system in one deployment, with four detections, all benign, so the field record contains no adversarial case. A single reviewer, the first author, adjudicated all four; a reviewer less familiar with the system might behave differently, in either direction. The justifications were recorded as free text and categorized by the authors after the fact. Rendered presentation has not yet been tested against string presentation, and the completeness figures are reported as logged, without independent reconciliation.

CONCLUSION

A PDF is read twice, by a person who sees the rendered page and by an extractor that sees the data layer, and the readings differ in both directions. SourcePoint meets silent loss with a control that decides without asking, and across two batches it withheld 13 documents over 51 tokens rather than emit a package that was quietly short. It met concealment with the conventional control, a detection escalated to a person, and all four detections were released,

one on the stated ground that the text could not be seen. The completeness control had the evidence its decision required; the conventional aid passed its decision to a person and withheld the evidence, showing a string where the question was about geometry. The recommendations follow: present the concealment rather than the content, ask whether it is explicable rather than whether the content is safe, and make release the action that needs a reason. Until they are tested, the field record stands: four justified releases that proved correct, and nothing the reviewer was shown that could have told a correct release from a mistaken one.

ACKNOWLEDGMENT

The authors thank ElevateEdAZ and ARCWERX for contributing reviewers to SourcePoint's testing.

Generative AI disclosure: Claude (Anthropic) assisted with drafting this manuscript from the authors' abstract, case study, and run records, with consolidating that material into the conference format, and with reviewing the draft for internal consistency. The system, field observations, study design, and technical content are the authors' own, and the authors reviewed and take responsibility for the final text.

REFERENCES

Bainbridge, L. (1983). Ironies of automation. Automatica, 19(6), pp. 775–779.

Elish, M.C. (2019). Moral crumple zones: cautionary tales in human-robot interaction. Engaging Science, Technology, and Society, 5, pp. 40–60.

Greshake, K., Abdelnabi, S., Mishra, S., Endres, C., Holz, T. and Fritz, M. (2023). Not what you've signed up for: compromising real-world LLM-integrated applications with indirect prompt injection. In: Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security (AISec '23). New York: ACM, pp. 79–90.

ISO (2008). ISO 32000-1:2008 Document management. Portable document format. Part 1: PDF 1.7. Geneva: International Organization for Standardization.

Lee, J.D. and See, K.A. (2004). Trust in automation: designing for appropriate reliance. Human Factors, 46(1), pp. 50–80.

NIST (2023). Artificial intelligence risk management framework (AI RMF 1.0). NIST AI 100-1. Gaithersburg, MD: National Institute of Standards and Technology.

OWASP (2025). OWASP top 10 for LLM applications 2025. OWASP GenAI Security Project. Available at: https://genai.owasp.org/llm-top-10/

Parasuraman, R. and Manzey, D.H. (2010). Complacency and bias in human use of automation: an attentional integration. Human Factors, 52(3), pp. 381–410.

Parasuraman, R., Sheridan, T.B. and Wickens, C.D. (2000). A model for types and levels of human interaction with automation. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, 30(3), pp. 286–297.

S2RC (2026). SourcePoint case study: provably complete. Phoenix, AZ: Stone Solutions + Research Collective LLC.

Skitka, L.J., Mosier, K.L. and Burdick, M. (1999). Does automation bias decision-making? International Journal of Human-Computer Studies, 51(5), pp. 991–1006.

Sorkin, R.D. and Woods, D.D. (1985). Systems with human monitors: A signal detection analysis. Human-Computer Interaction, 1(1), pp. 49–75.

Sugiyama, S. and Eguchi, R. (2025). 'Positive review only': researchers hide AI prompts in papers. Nikkei Asia, 1 July. Available at: https://asia.nikkei.com/business/technology/artificial-intelligence/positive-review-only-researchers-hide-ai-prompts-in-papers

van der Sijs, H., Aarts, J., Vulto, A. and Berg, M. (2006). Overriding of drug safety alerts in computerized physician order entry. Journal of the American Medical Informatics Association, 13(2), pp. 138–147.

Woods, D.D. (1995). The alarm problem and directed attention in dynamic fault management. Ergonomics, 38(11), pp. 2371–2393.