MatCurvs: Article Real-Coordinate Curve Extraction for Agent-Ready Materials Reasoning

Liang Yin[1,2] yinliang22@mails.ucas.ac.cn
Zhan'ao Yao[1] yaozhanao@mail.sic.ac.cn
Jiahui Shi[3,*] jiahui.shi@ia.ac.cn
Songlin Yu[1,4,*] yusonglin22@mails.ucas.ac.cn
Jianjun Liu[1,2,5,*] jliu@mail.sic.ac.cn
[1]Shanghai Institute of Ceramics, Chinese Academy of Sciences;
[2]School of Advanced Interdisciplinary Sciences,
University of Chinese Academy of Sciences;
[3]Institute of Automation, Chinese Academy of Sciences;
[4]Mattok (Shanghai) AI Technology Co., Ltd.;
[5]Hangzhou Institute for Advanced Study,
University of Chinese Academy of Sciences
[*]Corresponding authors: Jiahui Shi, Songlin Yu, and Jianjun Liu.

Abstract

Current vision–language models struggle to extract precise values from materials-science line plots. We address this gap with three coupled artifacts for curve extraction and downstream quantitative reasoning. **MatCurvs** is a real-coordinate chart benchmark for materials-science spectra, organised in three tiers spanning XRD, Raman, XPS, XAFS, EIS, GCD and CV: $1{,}375{,}165$ classified subfigures from $55{,}763$ articles (L0); 204 panels with manual per-curve pixel polylines (L1); and $8{,}026$ screened real-coordinate curves over $3{,}817$ panels (L2). **MatDeplot** is a local extractor that recovers 45.8% of curves at IoU$\geq$0.5 against L1 ground truth using a 5-pixel GT rasterisation, versus $\leq 1.2\%$ for the strongest hosted VLM — a $38\times$ pixel-anchoring gap, closed in roughly 1.5 seconds per image. **MatCurvs-Reasoning** is a $14{,}740$-question deterministic numerical evaluation: feeding MatDeplot's extracted (x, y) to an LLM cuts median relative error from 54% (image-only VLM) to about 4%, and a text LLM with the data alone matches the same VLM prompted with image and data together. By turning each published chart into a numerically queryable curve, the released artifacts let a deployer compute derived materials parameters — Raman $\mathrm{I}_D/\mathrm{I}_G$ disorder ratios, XRD Scherrer crystallite sizes, phase-purity audits — directly from a literature-scale chart corpus, where most papers under-report these numbers and the chart provides a record of the underlying measurements.

1 Introduction

Materials-science papers communicate quantitative measurements predominantly through line plots: XRD diffractograms with Miller-indexed peak positions, Raman spectra with $\mathrm{I}_D/\mathrm{I}_G$ ratios, EIS Nyquist plots with high-frequency Z' intercepts, GCD voltage–capacity curves, CV redox features, XAFS edge positions. Reading these panels quantitatively is the foundation on which downstream materials reasoning rests: peak positions and widths give phase identity and lattice parameters [1, 2], intensity ratios index disorder in carbon systems, and a recent line of spectra–structure–property work [3–5] shows that downstream conclusions — phase mapping, composition–performance correlation, mechanism inference — reduce to deterministic computation once the underlying (x, y)

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

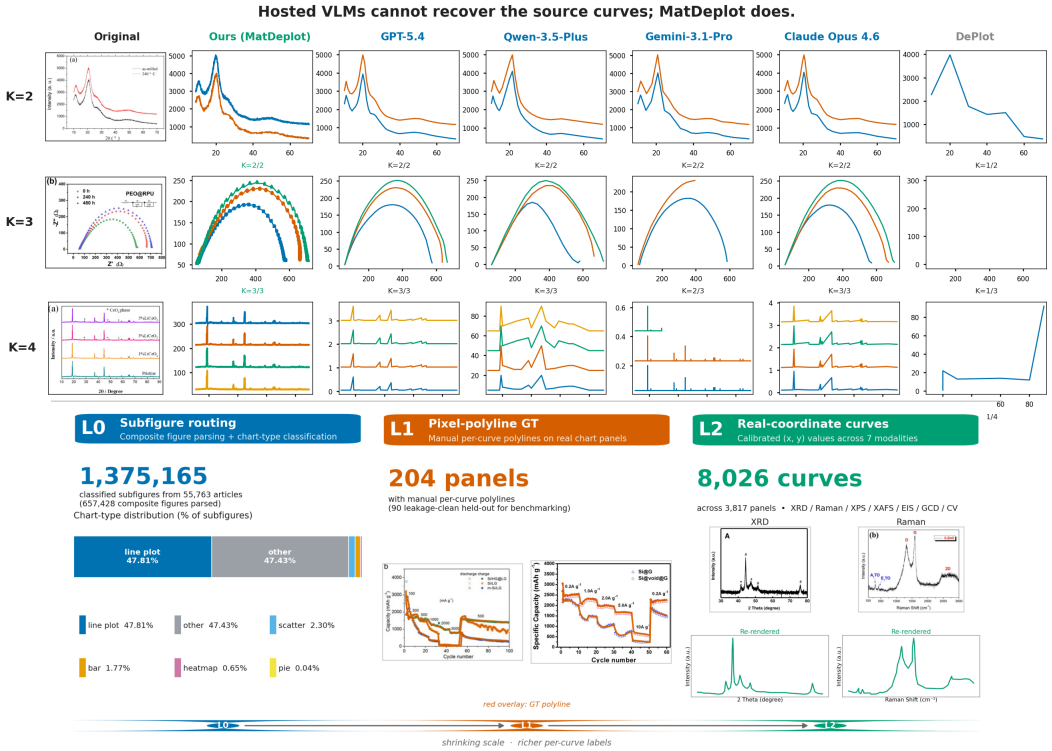

Figure 1: **MatDeplot recovers source-curve details missed by hosted VLMs. Top**: comparison on three L2 panels with curve counts $K \in \{2, 3, 4\}$ (XRD overlay, EIS Nyquist, and stacked XRD). Columns show *Original*, *Ours* (MatDeplot), four hosted VLMs, and DePlot [6]. The VLMs share one prompt, and all x-axes use the original range. Subcaptions report $K_{\text{pred}}/K_{\text{gt}}$; the full grid is in Figure F.2. **Bottom**: MatCurvs spans three tiers along the article-to-curve path. **L0** routes $1{,}375{,}165$ classified subfigures (47.81% line plots). **L1** provides manual per-curve pixel polylines for 204 panels, including an $n{=}90$ leakage-clean holdout for evaluating pixel-faithfulness. **L2** supports agentic reasoning with $8{,}026$ screened real-coordinate curves from $3{,}817$ panels across XRD, Raman, XPS, XAFS, EIS, GCD, and CV.

values are recovered as a numerical array rather than a rendered image. Recovering those values is therefore the gating step, and it is the step at which current literature-scale pipelines fail.

Existing literature-mining [7–10] and document-parsing systems [11–13] read text, equations and tables well, but largely stop at the subfigure boundary: the per-curve numerical content inside line plots remains unprocessed. The default interface for reading these panels is now to hand the chart image to a hosted vision–language model [14–16] and ask a natural-language question [17–20]. Precision matters even in comparatively simple published plots. For example, the Co@CoO XRD pattern of Yan et al. [21] comes from a crystalline metallic-Co core with an amorphous CoO shell; its diffraction peaks arise from cubic and hexagonal Co phases. In the rendered 693×585-pixel panel, approximately 593 horizontal plotting intervals span 30–80° in 2θ. A displacement of one to three pixels therefore corresponds to 0.084–0.253°, or 0.17–0.51% of the axis span. This sensitivity motivates explicit coordinate extraction before calculating materials parameters.

At scale, against manual L1 pixel polylines, hosted VLMs reach at most $\mathbf{1.2}\%$ at the $\text{LR}_{0.5}$ operating point, whose GT polylines are rasterised at 5 pixels, vendor-uniform across four 2026-Q2 frontier families [22–25]. The reason is that the chart's numerical content is never lifted into the model's reasoning context, and the existing benchmark literature does not catch this gap: chart-question-answering benchmarks [26–28] score question-answering *about* charts [29–33], and the broader multimodal-evaluation suites [34–36] measure visual-reasoning competence in general, neither targeting numerical extraction from dense scientific spectra. The chart-derendering line [6, 37–39] is

trained on tabular charts (bar, pie, short line lists) [40–43] and truncates after ~ 10 (x, y) pairs per series, recovering zero polylines on dense XRD or Raman spectra (Table 2). Prior local foreground-segmentation baselines [44–46] likewise collapse on dense-scatter plots. None measure pixel-level extraction over scientific spectra, and the substrate an agent [47, 48] can interpolate, integrate, differentiate, or compare across papers simply does not exist yet.

Our solution and contributions. We address this with three coupled artifacts. **First**, we release **MatCurvs**, a three-tier real-coordinate chart benchmark for materials-science spectra: **(L0)** $1{,}375{,}165$ classified subfigures from $55{,}763$ articles, organised by chart-type and a 3×10 characterisation taxonomy (three first-level regimes — Experimental, Computational, Schematic — and ten canonical second-level modalities including Performance, Microscopy, Structural Characterisation, Compositional, Conceptual, Structural Diagrams, Process Flow, DFT, MD and FEM; full listing in Appendix B); **(L1)** 204 panels with manual per-curve pixel polylines plus an $n{=}90$ SHA-disjoint leakage-clean hold and the curve-level line-recall metric LR_θ; **(L2)** $8{,}026$ screened real-coordinate curves over $3{,}817$ panels spanning XRD, Raman, XPS, XAFS, EIS, GCD and CV (Section 4.3). **Second**, we release **MatDeplot**, a local extraction pipeline that achieves a $\mathbf{38}\times$ improvement over the strongest hosted VLM in pixel-anchored curve recovery, at ~ 1.5 s/image. **Third**, we release **MatCurvs-Reasoning**, a $14{,}740$-question deterministic numerical evaluation: feeding MatDeplot's extracted (x, y) to the same VLM cuts median relative error from roughly 54% (image-only) to 5.5% (image + data), and an image-blind text LLM operating on the data alone reaches 4.2% — evidence that extraction recovers the reasoning-relevant numerical content of the chart, and a small Agent-Ready demonstration that on $n{=}7$ published papers reaches $19/19$ on derived-parameter tasks where an image-only VLM scores $2/19$ (Section 7).

Data and code. Dataset: `https://huggingface.co/datasets/yinliang22/matcurvs`.
Code: `https://github.com/yinliang420/matcurvs`.

Section 2 situates this work; Sections 3 and 4 formalise the problem and describe MatCurvs; Section 5 details MatDeplot; Sections 6 and 7 report the pixel-anchor and agentic-reasoning headlines; Section 8 catalogues limitations.

2 Related Work

Chart derendering and chart-VQA benchmarks. Chart-to-table models [6, 37–43, 49] and visual document-understanding pretrains targeting charts as a sub-task [12, 13, 50] target tabular charts (bar, pie, short line lists) and produce zero recoverable polylines on dense XRD/Raman spectra (Table 2). Of this line, we benchmark DePlot and OneChart as representative open-weight chart-derenderers on L1; the remaining citations (MatCha, UniChart, ChartAssistant, ChartLLama, StructChart, ChartGemma) share the same tabular-pretrain limitation and a head-to-head on dense scientific spectra is left as future work. Existing chart-VQA benchmarks [26–33] and the broader multimodal-evaluation suites [34–36] score *question-answering about* charts rather than *numerical extraction from* them; MatCurvs provides a chart benchmark with per-curve pixel *and* real-coordinate annotations over materials-science chart families.

Hosted VLMs and dense segmentation. Frontier hosted VLMs [14–17] and open-weight multimodal stacks [18–20] match a curve's shape in prose but cannot anchor to source pixels (§6). MatDeplot's foreground head is a Mask2Former [51, 52] on a MiT-B3 encoder [53] with $q{=}12$ object queries, replacing the DeepLabV3+/UNet++ [44, 46] on ResNeSt [45] backbones used by prior local chart-FG baselines, which we find dominant-mask-biased on dense scatter (Table 2). Class-agnostic universal-segmentation foundations [54, 55] are an alternative but were not trained for line-curve foreground.

Scientific literature mining and agent loops. WebPlotDigitizer [56] digitises curves by hand; ChemDataExtractor [7] and the text-mining pipelines of Olivetti et al. [8], Kononova et al. [9], Tshitoyan et al. [10], Fang et al. [11] ingest text, equations and tables but not figures, and the recent line of spectra-to-property automation work [1–5] consumes already-digitised spectra rather than reading them off published charts; materials- informatics agents that query structured databases [57, 58] or assemble ReAct/Toolformer [47, 48] and autonomous chemistry loops [59, 60] need a reading layer with per-call latency and abstain semantics — the regime our JSON output is designed for.

Table 1: The three tiers of the MatCurvs benchmark.

Tier	N	Annotation or screening	Used in
L0	1,375,165	chart-type $\times$ 3×10 characterisation taxonomy	§4.1, App. B
L1	204 (90 hold)	manual per-curve pixel polylines, 5-px raster	§6
L2	8,026 curves / 3,817 panels	VLM-screened real-coordinate curves	§7

3 Problem Formalization

Article-scoped inputs and outputs. The natural unit of scientific reasoning is the *subfigure*: a materials-science figure is a multi-panel composite (synthesis schematic + XRD pattern + EIS Nyquist + GCD curves), and any cross-paper agent has to first isolate, classify, then read each panel's curves. The MatCurvs problem is therefore article-scoped: input is a PDF $\mathcal{P}$; output, per figure in $\mathcal{P}$, is **(i)** subfigure bounding boxes, **(ii)** a chart-type label plus a 3×10 characterisation tag (XRD, XPS, GCD; experimental / computational / schematic), and **(iii)** for line panels, for each curve $i \in \{1, \ldots, K\}$ both the pixel set $\hat{C}_i$ and the real-coordinate sequence $\hat{S}_i{=}((x_1^i, y_1^i), \ldots, (x_{n_i}^i, y_{n_i}^i))$ from axis calibration, with a per-line confidence and abstain flag. Reference targets: L1's manual pixel polyline C_j^* (rasterised at 5 px) and L2's VLM-screened real-coordinate sequence S_j^*.

Pixel anchoring: line recall LR_θ. Given prediction $\{\hat{C}_i\}$ and L1 ground truth $\{C_j^*\}$, we compute pairwise IoU, solve the Hungarian assignment $\sigma(\cdot)$, and define

$$\mathrm{LR}_\theta \;=\; \frac{1}{K}\sum_{j=1}^{K} \mathbb{1}\Big[\mathrm{IoU}(C_j^*, \hat{C}_{\sigma(j)}) \geq \theta\Big], \tag{1}$$

the fraction of GT curves whose matched prediction exceeds threshold θ at the dataset's 5-px rasterisation. We report $\theta \in \{0.5, 0.8\}$ as the principal headline metric.

Permissiveness hierarchy and real-coordinate agreement. Three nested agreement levels let a reader claim a chart was "read": **Shape** (Inst-F1 / DTW, no pixel anchoring), **Instance** (right K, right colour-to-curve assignment), and **Pixel** (IoU$\geq\theta$ on a 5-px rasterisation), with Pixel $\subset$ Instance $\subset$ Shape as classifier strength. Hosted VLMs occupy Shape-only; real-coordinate reading requires Pixel *plus* axis calibration. Beyond pixel anchoring, we measure real-coordinate agreement $\mathrm{DTW}_\Delta(\hat{S}_i, S_i^*)$ in physical units normalised by axis range — the metric downstream reasoning depends on.

4 The MatCurvs Benchmark

MatCurvs is structured in three tiers, each with a distinct role (Table 1). **L0** is a routing layer: every subfigure in our 55,763-article corpus is classified by chart type and characterisation taxonomy, so that downstream consumers can filter line plots from microscopy or computational panels. **L1** is a small ($n{=}204$), manually pixel-polylined subset that supports the rigorous LR_θ measurement of pixel anchoring. **L2** is a large ($\approx 8{\times}10^3$ curves over 3,817 panels), semi-automatic, VLM-judge-screened set of real-coordinate curves that feeds the downstream agentic-reasoning evaluation in Section 7. L1 and L2 are coupled: agreement between L1's manual polylines and MatDeplot's reduction to real coordinates on the same images provides a cross-tier diagnostic, rather than independent pointwise validation of the L2 corpus. The screening-yield and verifier-sensitivity boundaries are reported in Appendix P.

4.1 L0 — Subfigure routing

Each article's figures are first segmented into individual subfigures by an in-house YOLOv11x detector [61] trained on 1,366 manually annotated panels. On a held-out test split the detector reaches $\mathrm{mAP}_{0.5}{=}0.984$, $\mathrm{mAP}_{0.5:0.95}{=}0.932$, precision 0.967 and recall 0.962, outperforming Mask-RCNN-R50/R101 (0.775 / 0.785 on $\mathrm{mAP}_{0.5:0.95}$) and a 7B-parameter Qwen-VL baseline (0.742); see Appendix B (Figure B.1, Figure B.2). Each cropped subfigure is then routed by a multimodal-LLM

classifier (qwen-vl-max with a chain-of-thought prompt; Appendix B.2) into one of 6 chart types and a 3×10 characterisation taxonomy. Across the 55,763-article corpus this yields $\mathbf{1{,}375{,}165}$ classified subfigures, of which $\mathbf{47.81}\%$ (657,428) are line plots; per-class classifier accuracy on a stratified audit (1,000 subfigures per class) is 90.6%–97.1% (mean 94.3%; per-class counts and correct/failed bars in Figure B.3–Figure B.6).

4.2 L1 — Pixel-polyline ground truth

L1 is a 204-panel subset of L0 with manual per-curve pixel polylines. For each panel, an annotator traces every visible data curve in the chart with a polyline at the original image resolution; the raw vertices are stored alongside an axis calibration (pixel-to-data-unit affine) so that each curve is available both as (u, v) pixels and as (x, y) in physical units. At evaluation time the polylines are rasterised at thickness 5 px following published GT conventions, yielding the binary mask sets C_j^* used by LR_θ (Equation (1)).

Annotation, QC, and leakage-clean hold (n=90). A single trained annotator marked all 204 panels using a WebPlotDigitizer-derived tool; a second annotator independently re-annotated 20 panels for inter-annotator consistency, and a domain expert reviewed every panel. The MatDeplot Mask2Former segmenter is trained on a 182-panel real-chart corpus that overlaps L1 by SHA-256: 114 of the 204 L1 panels appear in the training root and 90 do not. We define the n=90 **leakage-clean hold** as the disjoint $\{204\} \setminus \{\text{train}\}$ complement and report it as the headline split for LR_θ throughout §6; the contaminated n=204 aggregate is reported alongside as an upper-bound reference. Audit procedure and the per-pair SHA list are in Appendix L.

Dual purpose. L1 supplies the gold-standard pixel polylines for the LR_θ headline of §6, and on the 49-panel intersection L1 ∩ L2 acts as an external calibration validating the L2 curation protocol (§4.3).

4.3 L2 — Screened real-coordinate curves

L2 contains 8,026 **real-coordinate curves over** 3,817 **panels** drawn from 2,344 articles, spanning Raman, XAFS, XRD, GCD, CV, EIS and XPS. Each curve is provided as an ordered $(x_{\mathrm{real}}, y_{\mathrm{real}})$ sequence in physical units, with the underlying chart image preserved alongside.

Curation protocol and cross-tier validation. Each candidate panel runs through MatDeplot (§5) to produce a draft extraction with three confidence signals (Mask2Former FG fraction, k-means inertia, per-curve abstain). A qwen-vl-max judge inspects the original image side-by-side with a clean re-rendering and applies a four-criterion check (curve count, shape, axis range, broken segments); only all-criterion-pass panels enter L2. A four-judge $\geq$ 3-of-4 majority-fail consensus (qwen-vl-max, GPT-5.4, Gemini-3.1-pro, Claude Opus 4.6; pairwise $\kappa \in [0.05, 0.27]$, full table in Tables M.2 and M.4) abstains on 64/90 L1-hold panels and yields $\Delta\,\mathrm{LR}_{0.5}$=+0.455 between kept and abstained subsets, anchoring the single-judge protocol. L2 ships both verdicts so downstream users can re-curate under any threshold; verifier sensitivity and the distinction between screening and pointwise labels are detailed in Appendix P. On the 49 L1 ∩ L2 panels, the median DTW_Δ residual is dominated by curve-label ambiguity in multi-curve panels (§8), not extraction noise.

5 MatDeplot: The Extraction Method

MatDeplot is the upstream extraction *method* that produces the L2 layer of MatCurvs (the two contributions are distinct: MatCurvs is the dataset, MatDeplot is one strong baseline that populates L2). A document is processed in three stages (PDF → MinerU subfigure parsing → YOLOv11x [61] detector → chart-type classifier); the line-panel reader of Stage 3 is the focus of this section. The full document-scale schematic is in Figure C.1 (App. C).

Stage 1 — Subfigure detection. PDFs are parsed by MinerU and panels cropped by a YOLOv11x detector ($\mathrm{mAP}_{50:95}$=0.932, vs. 0.775 for Mask R-CNN R-50; full benchmark in Appendix B, Figure B.1).

Table 2: **Comparison across extractor families on the L1 leakage-clean hold** (n=90 panels, manual pixel-polyline GT; absolute gap +44.6 pp, 38× at VLM near-zero floor). *DePlot/OneChart are tabular-chart pretrains, out-of-distribution on dense scientific spectra (reported as a "the chart-derendering line does not transfer" reference, not a fair comparator); modern open-weight derenderers (UniChart [39], ChartGemma [43]) trained on dense line plots remain a future-work comparator.

Paradigm	System	$\mathrm{LR}_{0.5}\uparrow$	Inst-F1↑	$/img↓	local?
Chart-derendering*	DePlot [6]	0.0%	0.058	local	yes
	OneChart [37]	0.0%	0.052	local	yes
Local FG seg	DeepLabV3+/ResNeSt [44, 45]	1.1%	0.137	local	yes
	UNet++/ResNeSt [45, 46]	0.4%	0.119	local	yes
Hosted VLM (E2E)	Qwen-3.5-plus	0.0%	0.837	$0.005	no
	gemini-3.1-pro-preview	0.4%	0.343	$0.012	no
	gpt-5.4	1.2%	0.915	$0.118	no
	claude-opus-4.6	0.5%	0.898	$0.080	no
Ours	**MatDeplot**	**45.8%**	**0.942**	$0.012	yes

Stage 2 — Subfigure classification. A multimodal-LLM classifier (qwen-vl-max with a chain-of-thought prompt) routes each cropped panel into one of 6 chart types × a 3×10 characterisation taxonomy at 94.3% mean per-class accuracy on 1,000 images per characterization category (90.6%–97.1% across families; Appendix B.3).

Stage 3 — Line-chart reading. Six cells convert one chart image into per-curve real-coordinate polylines with per-curve confidence and an explicit abstain flag, matching the schematic of Figure C.2.

Cells 1 & 2 — axis and legend. Two YOLOv11 detectors return tick pixel coordinates (used to fit a 2×3 affine $\mathcal{C}:\text{pixel}\to\text{data units}$) and a legend bounding box that is masked out before downstream clustering.

Cells 3 & 4 — foreground segmentation with fallback. A Mask2Former head on a MiT-B3 backbone with q=12 queries [51, 53] emits a soft foreground map $F \in [0,1]^{H\times W}$ (multi-scale TTA, flip-fused, binarised at θ_{fg}=0.25; architecture/data/loss sweep in Appendix D). A gate $g(F)$ on the foreground fraction $\rho(F)=|P|/(H\cdot W)$ routes pixels with $\rho(F)\geq\tau_{\text{fg}}$=0.003 to Cell 5; otherwise (scatter panels the FG head silently misses) Cell 4 falls back to clustering the non-Otsu-white plot interior, guaranteeing non-empty output and recording the fallback flag in the output record.

Cells 5 & 6 — curve separation and polyline extraction. Each foreground pixel is embedded as $\phi(u,v) = (L^*, a^*, b^*, w_{xy}u', w_{xy}v') \in \mathbb{R}^5$ with w_{xy}=0.5, and k-means [62, 63] is run at cluster count k from a cached Gemini-3.1-pro-preview call (exact match on 83.3% of n=90; Appendix K). Post-processing applies morph-close 5, 3×3 dilation, a 200-px min-area filter and an LAB-ΔE<12-gated dilation that recovers anti-aliased edges without colour bleed. Each cluster is then skeletonised into a single polyline, projected through $\mathcal{C}$ into real (x, y) coordinates, and tagged with confidence $c_i=\text{mean}_{C_i}F$; curves below c_i<0.30 are emitted with an abstain flag set, and the released JSON carries both the pixel polyline and the data-space series.

Comparison across extractor families. Table 2 pits MatDeplot (Stage 3) against the dominant alternative paradigms on the n=90 leakage-clean hold of the L1 pixel-polyline tier: chart-derendering models (DePlot [6], OneChart [37]), the strongest prior local foreground baselines (DeepLabV3+/ResNeSt [44, 45] and UNet++/ResNeSt [46] as published), and four hosted VLMs operated end-to-end (GPT-5.4, Claude Opus 4.6, Gemini-3.1-pro-preview, Qwen-3.5-plus). Headline pixel-anchoring metric is $\mathrm{LR}_{0.5}$; cost is per-image dollar at a 2026-Q2 snapshot of vendor prices.

The vendor-uniform pixel-anchoring failure of hosted VLMs ($\mathrm{LR}_{0.5}\leq1.2\%$) and the silent-collapse of the prior local segmenters ($\leq$ 1.1%) are the two effects the MatDeplot recipe is designed to remove. Architectural ablations (Mask2Former architecture/data/loss sweep, k-source variants, fallback trigger threshold, two-mode shape-vs-pixel toggle) are in Appendix D.

6 Closing the Pixel-Anchor Gap

We measure how far the gap between hosted VLMs and a local extraction pipeline can be closed when the goal is pixel-level faithfulness, using L1's manually pixel-polylined hold as the anchor.

Summary across extractor families. On the legacy 36-image test set (the only anchor where DePlot, OneChart, oracle-FG and hosted VLMs share an evaluator with us), MatDeplot reaches **DTW-Δ=14.7, CCR**=0.96**, Inst-F1**=0.52 — 9.6× lower DTW than the best learned-FG baseline (UNet++/ResNeSt 140.4), 4.3× lower than the best 2026-Q2 VLM (Qwen-3.5-Plus 63.0), at 1.5 s/image locally. *Shape-mode* wins these shape-regime metrics; the pixel-mode headline of the next paragraph uses a different Cell 5 post-processing tuned for LR_θ instead, with Table E.1 reporting both modes side-by-side. The oracle-FG breakdown is in Table E.2.

Faithful-reconstruction headline ($\mathrm{LR}_{\theta_{\mathrm{IoU}}}$). Table 3 reports $\mathrm{LR}_{\theta_{\mathrm{IoU}}}$ on the leakage-clean n=90 byte-disjoint complement of MatCurvs-204 (the FG model's training set overlaps $114/204$ images; Appendix L). The pixel mode records $\mathrm{LR}_{0.5}$=45.8% / $\mathrm{LR}_{0.6}$=35.5% / $\mathrm{LR}_{0.7}$=11.8% / $\mathrm{LR}_{0.8}$=0.5% ($1/203$ lines, i.e. strict pixel anchoring at IoU$\geq$0.8 remains essentially unsolved); on the contaminated n=204 aggregate the corresponding numbers are $54.7\%/38.7\%/14.7\%/1.1\%$. The substantive headline is $\mathrm{LR}_{0.5}$=45.8% on the clean hold: nearly half of all GT curves are pixel-localised at IoU $\geq$ 0.5, while every hosted VLM caps at $\mathrm{LR}_{0.5} \leq 0.012$. For the VLM rows we report the Curve-PQ-derived upper bound $\mathrm{LR}_\theta \leq \mathrm{PQ}/\theta$ (rasteriser-slack discussion in App. K); since PQ<0.01 for every VLM, every LR entry stays ≤ 0.013 at every threshold. Complementary coordinate-level analysis on the same clean hold finds matched support within ± 1 image column at $196{,}461/229{,}950$ annotated centerline locations (85.4%). The median vertical error at these supported locations is 2 pixels (0.47% of image height; Appendix N).

Table 3: Faithful-reconstruction $\mathrm{LR}_{\theta_{\mathrm{IoU}}}$ on the leakage-clean n=90 hold (MATDEPLOT pixel-mode recipe of Algorithm 1 with the cached Gemini-K source). $\mathrm{LR}_{\theta_{\mathrm{IoU}}}$ is the fraction of GT curves with pixel-IoU–matched IoU $\geq \theta_{\mathrm{IoU}}$ at thickness 5 px. Hosted-VLM rows quote the directly-measured Curve-PQ on the same n=90 hold; the implied upper bound $\mathrm{LR}_{\theta_{\mathrm{IoU}}} \leq \mathrm{PQ}/\theta_{\mathrm{IoU}}$ (panoptic-style; see App. K) is shown alongside, since raw VLM polylines were not cached. Per-method Inst-F1 is in the appendix (Table E.5).

Method (n=90)	$\mathrm{LR}_{0.5}\uparrow$	$\mathrm{LR}_{0.6}\uparrow$	$\mathrm{LR}_{0.7}\uparrow$	$\mathrm{LR}_{0.8}\uparrow$	Curve-PQ↑
MATDEPLOT (pixel)	**45.8%**	**35.5%**	**11.8%**	**0.5%**	**0.126**
GPT-5.4	$\leq 0.0\%^{\ddagger}$	$\leq 0.0\%^{\ddagger}$	$\leq 0.0\%^{\ddagger}$	$\leq 0.0\%^{\ddagger}$	0.000
Gemini-3.1-pro	$\leq 1.2\%^{\ddagger}$	$\leq 1.0\%^{\ddagger}$	$\leq 0.9\%^{\ddagger}$	$\leq 0.7\%^{\ddagger}$	0.006
Qwen-3.5-plus	$\leq 0.0\%^{\ddagger}$	$\leq 0.0\%^{\ddagger}$	$\leq 0.0\%^{\ddagger}$	$\leq 0.0\%^{\ddagger}$	0.000

‡ VLM upper bound via $\mathrm{LR}_\theta \leq \mathrm{PQ}/\theta$; the two rasterisers differ only by anti-aliased edge slack (App. K). Pixel-mode Curve-PQ (0.126) trails shape-mode (0.229, Table E.1); LR was the metric pixel mode was tuned for. Per-stratum metrics are in Table E.4.

Significance and breakdowns. The unambiguous lead is on LR_θ and DTW-Δ. The pixel-mode recipe is the median of a 14-candidate post-processing sweep on the same hold (IQR $\mathrm{LR}_{0.5} \in [44.6, 46.6]\%$); the fallback never fires at τ_{fg}=0.003 on this hold, and per-stratum / per-chart breakdowns are in Appendices E and G.

7 MatCurvs-Reasoning

Pixel-faithful chart reading is only useful insofar as it changes what an agent can *do* with the chart. We turn this into a controlled measurement. **MatCurvs-Reasoning (MR)** is a benchmark of 14,740 deterministic numerical-reasoning questions ($3{,}817$ panels $\times \sim 3.86$ questions/panel) constructed from L2 (§4.3). Each question has scipy/numpy-computable ground truth. We pose every question to four methods and measure the gap.

Question taxonomy. Each panel receives three Cat-A generic questions — $y(x_0)$ via interpolation, argmax over a sub-range, and the full-width-half-maximum of the most-prominent

peak (with *peak-to-peak amplitude* as a fallback when no peak is detectable). Multi-curve XRD/Raman/XPS/XAFS panels additionally receive a Cat-C *stacked-intensity-offset* task whose ground truth is the difference of bottom-quartile baselines between adjacent stacked curves. Raman panels receive two Cat-B family questions (top-3 peak positions and the I_D/I_G ratio); EIS panels receive the high-frequency Z' intercept.

Four-method comparison. **M1 (VLM image-only)** sends the chart image + the question to a VLM. **M2 (VLM image+data)** additionally injects the 500-point downsampled (x, y) data of every curve into the prompt. **M3 (LLM + data)** sends only the data and the question to a text LLM (no image). **M4 (scipy oracle)** runs the deterministic GT formula on the L2 (x, y) data and is reported on the rigorous set as the extraction-error ceiling against L1 manual GT.

Headline: rigorous set (n=49 panels, 188 questions, GT from L1 manual). Table G.1 reports the comparison across all four methods and three model families per VLM/LLM method. **The single change from M1 to M2/M3 reduces median relative error by 3.3–3.6×:** from M1's best $\mathrm{medRel} = 15.4\%$ (GPT-5.4 image-only) to M2's 4.3% (GPT-5.4 image+data) and M3's 4.6% (GPT-5.4 data-only). A text LLM operating on the extracted curves alone (M3) matches a VLM given image plus the same curves (M2): medRel=4.6% vs. 4.3%, with M3 even reaching the deterministic `scipy` oracle's 6.7% within noise. Since M2's input is a strict superset of M3's (image + data $\supset$ data), the parity is positive evidence that the chart image, beyond what MatDeplot has already extracted, contributes no additional load-bearing numerical information for these tasks — i.e., on these families MatDeplot's extraction is *empirically information-sufficient*: the image's marginal contribution beyond the extracted (x, y) is statistically indistinguishable from zero on three of five task families and *negative* on the C-Stack offset task. We conjecture but do not prove general information-completeness; the controlled test is restricted to the families enumerated above. The within-instrument-resolution pass rate ($\mathrm{relErr} < 5\%$) jumps from 37% to 52–53% in the same pivot.

Where extracted data shines: stacked spectra (Cat-C). Authors of XRD/Raman/XPS/XAFS papers vertically offset successive spectra by a constant intensity for visual clarity — a publication convention invisible-yet-measurable in the data. Table G.2 (last row): image-only VLMs miss the offset by 99.2% (qwen-vl-max), 67.7% (gemini-3.1-pro), and 23.1% (gpt-5.4) medRel; the same panels solved on the extracted data drop to 3.5% and 3.1% for GPT-5.4 in M2 and M3, respectively, and to 2.8% for Claude in M3; Qwen M2 reaches 62.3%. This is the strongest single piece of evidence that pixel-faithful extraction unlocks reasoning the image alone cannot support.

Image contribution beyond extraction is small. On the GPT-5.4 model present in both M2 and M3, per-task differences sit within ≤ 1.2 pp on every family (Table G.2): A1 y-at-x_0 M2 5.0% vs. M3 6.2%, A2 argmax 0.4% vs. 0.5%, A3 FWHM 20.3% vs. 20.5%, B-Raman peaks 2.9% vs. 3.3%, and **C-Stack stacked-offset** 3.5% **vs.** 3.1% **where M3 is ahead** — the constant intensity offset that authors add for visual clarity is visible in the image but not pixel-quantifiable, and the data-only path actively gains from removing it. The image's marginal contribution beyond MatDeplot's extracted (x, y) is therefore at most a few percentage points on these reasoning families and negative on the publication-convention task. We claim empirical information-sufficiency for the quantitative-reasoning families measured here, not for all uses of the image (legend identity, panel labelling, schematic interpretation are out of the MR taxonomy).

Generalisation across families, scale, and task regimes. The same M1→M2/M3 drop reproduces across XRD (n=90), Raman (n=53) and XPS (n=34) panels (App. G), and at scale on the full 3,817-panel pool (14,740 questions: qwen-vl-max image-only medRel=54.3% → M2 5.5% → M3 qwen3-max 4.2%, pass-rate 15% → 29% → 50.5%). The breakdown cleaves into three regimes: direct numerical readout (A1, A2, B-EIS) is fully unlocked by extraction (99.8–100% → 0–2%); structured numerical reasoning (C-Stack offset) is unlocked by extraction plus a single `numpy.median` call (GPT-5.4: 23.1% → 3.5% in M2 and 3.1% in M3); and peak-detection-bounded tasks (A3 FWHM, B-Raman peaks) are extraction-quality bound, the `scipy.signal.find_peaks` oracle itself reaching only ∼28% medRel on A3 against L1 manual GT — the right target for future MatDeplot improvement.

Table 4: **Agent-Ready demonstration on materials-science derived parameters** (n=7 papers, 19 eligible cells across three tasks). T1 is on the carbon-Raman line ($5/7$ papers eligible); T2 and T3 are on the primary XRD line ($7/7$ papers eligible). VLM = gpt-5.4 with the relevant chart image only, T=0. Tolerances: 20% rel. for T1, 30% rel. for T2, ±1 in matched count for T3.

Task	MD hits	VLM hits	medAE (MD)	medAE (VLM)
T1 Raman I_D/I_G ratio	$5/5$	$2/5$	0.000	0.257 (ratio)
T2 XRD Scherrer D (nm)	$7/7$	$0/7$	0.000	15.0 nm
T3 XRD phase presence (count)	$7/7$	$0/7$	0.000	5.0 peaks
Aggregate	**19/19 (100.0%)**	**2/19 (10.5%)**	+89.5 pp	

The fixed-denominator propagation audit, calibration policy, and zero-shot dense-model control are reported in Appendices O and P.

Agent-Ready demonstration: materials-science derived parameters. We test three real materials-science scenarios where the answer is a derived parameter that published papers rarely print on the chart and that VLMs cannot extract from the image alone: (T1) the Raman I_D/I_G integrated-area ratio that quantifies carbon-framework disorder; (T2) the XRD Scherrer crystallite size $D{=}K\lambda/(\beta\cos\theta)$ in nm with Cu-Kα (λ=1.5406 Å, K=0.94); and (T3) an XRD phase-presence checklist that audits how many literature-canonical Miller-indexed reflections are actually present above noise. For each task the agent loads the per-line MatDeplot `.xlsx`, runs a domain-standard `scipy` recipe (trapezoidal area in a $\pm60\,\mathrm{cm}^{-1}$ window for T1; Lorentzian fit with sample-spacing-guarded half-max fallback for T2; $\pm0.5^\circ$ presence test against the JSON peak list at 5% amplitude threshold for T3), and reports a single number. The same question is posed to gpt-5.4 (T=0, one image per call) given only the relevant chart image. Tolerances are predeclared: 20% relative for T1, 30% relative for T2, ±1 in matched count for T3. The same n=7 battery / energy-storage papers as the Run 1 phase-identification cohort are used; selection by data-quality criteria only, with no inclusion/exclusion on model performance.

Result. MatDeplot+agent answers $19/19$ eligible cells within tolerance; image-only gpt-5.4 hits $2/19$, an **89.5 pp gap** (Table 4). T1 is the tightest task: the VLM produces ball-park values (0.8–1.4 for I_D/I_G) but 3/5 fall outside the 20% band. T2 and T3 are dominated by order-of-magnitude misses: on T2 the VLM returns $\{2.7, 3.7, 14, 22, 35, 48, 106\}$ nm against recipe-truths $\{1.16, 3.91, 7.09, 10.63, 14.55, 19.88, 3.42\}$ nm (it cannot estimate FWHM from pixels nor evaluate the Scherrer formula), and on T3 it reports $\{5, 6, 9, 12, 14, 22, 31\}$ peaks while the present count is $\{1, 3, 4, 5, 9, 9, 16\}$ (it counts visible peaks across all curves rather than checking presence at specific 2θ values). MatDeplot is $7/7$ on T2 and T3 deterministically because the recipes operate on the extracted (x, y) array.

8 Limitations

(a) Scope. The benchmark is built around materials-science spectra (XRD, Raman, XPS, XAFS, EIS, GCD, CV); the information-sufficiency claim is restricted to the quantitative reasoning families we test, not to chart understanding in general (legend identity, panel-level captioning, schematic reasoning, and the categorical phase-ID regime where chart-text OCR dominates are out of scope, see §7 and per-family detail in Appendix G). **(b) Pixel anchoring is partial.** MatDeplot reaches $\mathrm{LR}_{0.5}$=45.8% but only $1/203$ lines at the strict $\mathrm{LR}_{0.8}$ bar; matching WebPlotDigitizer-grade manual tracing remains open, and our throughput/quality trade-off (80–200× faster at lower fidelity) is the lever future work should push. **(c) Model choices and calibration.** The headline recipe uses cached Gemini-3.1-pro-preview curve counts; local Qwen2.5-VL and deterministic silhouette-k alternatives are evaluated in Appendix P. L2 screening uses qwen-vl-max, which is also evaluated in M1; cross-verifier comparisons are reported in Appendices M and P. The pipeline assumes locally interpretable panels, while shared axes or legends may require context beyond the crop. Absolute physical-unit output requires numeric tick calibration; otherwise, the strict policy abstains.

9 Conclusion

Quantitative materials-science evidence lives in line plots, but the default hosted-VLM interface fails to lift it into reasoning context: across four 2026-Q2 frontier families, image-only chart reading caps at $\mathrm{LR}_{0.5} \leq 1.2\%$ on a leakage-clean hold and misses 15–54% relative on a $14{,}740$-question numerical-readout suite. MatCurvs measures this gap precisely; MatDeplot, a local extractor at ~ 1.5 s/image with cached curve counts, closes most of it (45.8%, a $38\times$ improvement). Feeding the extracted (x, y) to an LLM cuts median relative error to 4–5%, an image-blind text LLM on the data alone matches a VLM given image plus data, and an Agent-Ready demonstration on $n{=}7$ published papers reaches $19/19$ vs. image-only $2/19$ — evidence that pixel-faithful extraction is a load-bearing transformation, not an auxiliary preprocessing step.

We release the benchmark (L0 routing, L1 manual polylines, L2 screened real-coordinate curves), MatDeplot's pipeline and checkpoint, the four-method evaluator, and the Agent-Ready demonstration under permissive licences. Treating extraction as load-bearing makes literature-scale materials meta-analysis tractable for downstream agents: a deployer can ingest tens of thousands of published spectra, recover their (x, y) traces locally, and feed them to a reasoning agent whose input is now a faithful numerical record rather than a visually parsed estimate. The methodology — pixel-polyline grounding, real-coordinate verification, and a controlled image-vs-data reasoning probe — transfers to any domain whose published findings depend on numbers buried inside charts. Open problems (strict-IoU pixel anchoring and reproducibility under closed-vendor k-sourcing, §8) frame the natural next steps.

Funding Statement

The AI-driven experiments, simulations, and model training were performed on the robotic AI-Scientist platform of the Chinese Academy of Sciences. The authors gratefully acknowledge financial support from the National Key R&D Program of China (2025YFF0516300), the National Science and Technology Major Project (2026ZD0124900), the National Natural Science Foundation of China (NSFC) (52541102, 22133005, U25A20229, 22403102), the Science and Technology Commission of Shanghai Municipality (LJ2024049, 23ZR1472600, 25CL2902100), the Youth Innovation Promotion Association of CAS (2022251), the Open Research Fund of Suzhou Laboratory (No. SZLAB-1508-2024-ZD018), and the Young Scientists Fund of The State Key Laboratory of Multimodal Artificial Intelligence Systems (E6S9020703).

Competing Interests Disclosure

The authors declare no competing interests.

References

[1] G. Guo, T. L. Saidi, M. W. Terban, et al. Ab initio structure solutions from nanocrystalline powder diffraction data via diffusion models. *Nature Materials*, 2025.

[2] Bin Cao, Qian Zhang, Zhenjie Feng, Taolue Zhang, Jiaqiang Huang, Lu-Tao Weng, and Tong-Yi Zhang. AI-driven structure refinement of X-ray diffraction. *arXiv preprint arXiv:2602.16372*, 2026. URL `https://arxiv.org/abs/2602.16372`.

[3] Shuo Feng, Meng Huang, Yanbo Li, Aoran Cai, Xiaoyu Yue, Song Wang, Linjiang Chen, Jun Jiang, and Yi Luo. Intelligent understanding of spectra: From structural elucidation to property design. *Chemical Society Reviews*, 54:8243–8286, 2025. doi: 10.1039/D4CS01293C.

[4] Yongqi Jin, Jun-Jie Wang, Fanjie Xu, et al. NMR-Solver: Automated structure elucidation via large-scale spectral matching and physics-guided fragment optimization. *Nature Communications*, 2026. doi: 10.1038/s41467-026-71315-0.

[5] R. Odobesku, K. Romanova, S. Mirzaeva, et al. Agent-based multimodal information extraction for nanomaterials. *npj Computational Materials*, 11:194, 2025. doi: 10.1038/s41524-025-01674-7.

[6] Fangyu Liu, Julian Martin Eisenschlos, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Wenhu Chen, Nigel Collier, and Yasemin Altun. DePlot: One-shot visual language reasoning by plot-to-table translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, 2023.

[7] Matthew C. Swain and Jacqueline M. Cole. ChemDataExtractor: A toolkit for automated extraction of chemical information from the scientific literature. *Journal of Chemical Information and Modeling*, 56(10):1894–1904, 2016.

[8] Elsa A. Olivetti, Jacqueline M. Cole, Edward Kim, Olga Kononova, Gerbrand Ceder, Thomas Yong-Jin Han, and Anna M. Hiszpanski. Data-driven materials research enabled by natural language processing and information extraction. *Applied Physics Reviews*, 7(4):041317, 2020.

[9] Olga Kononova, Haoyan Huo, Tanjin He, Ziqin Rong, Tiago Botari, Wenhao Sun, Vahe Tshitoyan, and Gerbrand Ceder. Text-mined dataset of inorganic materials synthesis recipes. *Scientific Data*, 6(1):203, 2019.

[10] Vahe Tshitoyan, John Dagdelen, Leigh Weston, Alexander Dunn, Ziqin Rong, Olga Kononova, Kristin A. Persson, Gerbrand Ceder, and Anubhav Jain. Unsupervised word embeddings capture latent knowledge from materials science literature. *Nature*, 571:95–98, 2019.

[11] Xi Fang, Haoyi Tao, Shuwen Yang, Chaozheng Huang, Suyang Zhong, Haocheng Lu, Han Lyu, Junjie Wang, Xinyu Li, Linfeng Zhang, and Guolin Ke. Uni-Parser technical report. *arXiv preprint arXiv:2512.15098*, 2025. URL `https://arxiv.org/abs/2512.15098`.

[12] Kenton Lee, Mandar Joshi, Iulia Turc, Hexiang Hu, Fangyu Liu, Julian Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2Struct: Screenshot parsing as pretraining for visual language understanding. In *International Conference on Machine Learning (ICML)*, 2023.

[13] Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Nam, Jinyoung Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. OCR-free document understanding transformer. In *European Conference on Computer Vision (ECCV)*, 2022.

[14] OpenAI. GPT-4V(ision) system card. Technical report, OpenAI, 2023.

[15] Anthropic. The Claude 3 model family: Opus, sonnet, haiku. Technical report, Anthropic, 2024.

[16] Gemini Team, Google. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[17] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.

[18] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[19] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[20] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

[21] Xiaodong Yan, Yuan Liu, Jinle Lan, Yunhua Yu, James Murowchick, Xiaoping Yang, and Zhonghua Peng. Crystalline–amorphous Co@CoO core–shell heterostructures for efficient electro-oxidation of hydrazine. *Materials Chemistry Frontiers*, 2(1):96–101, 2018. doi: 10.1039/C7QM00401J.

[22] OpenAI. Introducing GPT-5.4. OpenAI. https://openai.com/index/introducing-gpt-5-4/, 2026. Closed-source hosted model; queried via the vendor's documented Chat Completions endpoint in 2026-Q1/Q2.

[23] Anthropic. Introducing Claude Opus 4.6. Anthropic. https://www.anthropic.com/news/claude-opus-4-6, 2026. Closed-source hosted model; queried via the vendor's documented Messages endpoint in 2026-Q1/Q2.

[24] Google DeepMind. Gemini 3.1 Pro model card. Google DeepMind. https://deepmind.google/models/model-cards/gemini-3-1-pro/, 2026. Closed-source hosted model; queried via the vendor's documented `generateContent` endpoint in 2026-Q2.

[25] Alibaba Cloud. qwen3.5-plus model info. Alibaba Cloud. https://www.alibabacloud.com/help/en/model-studio/qwen3-5-plus, 2026. Closed-source hosted model; queried via the vendor's documented OpenAI-compatible Chat Completions endpoint in 2026-Q2.

[26] Samira Ebrahimi Kahou, Vincent Michalski, Adam Atkinson, Ákos Kádár, Adam Trischler, and Yoshua Bengio. FigureQA: An annotated figure dataset for visual reasoning. In *International Conference on Learning Representations (ICLR) Workshop Track*, 2018.

[27] Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. DVQA: Understanding data visualizations via question answering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5648–5656, 2018.

[28] Ritwick Chaudhry, Sumit Shekhar, Utkarsh Gupta, Pranav Maneriker, Prann Bansal, and Ajay Joshi. LEAF-QA: Locate, encode & attend for figure question answering. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3512–3521, 2020.

[29] Nitesh Methani, Pritha Ganguly, Mitesh M. Khapra, and Pratyush Kumar. PlotQA: Reasoning over scientific plots. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1527–1536, 2020.

[30] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, 2022.

[31] Zhengzhuo Xu, Sinan Du, Yiyan Qi, Chengjin Xu, Chun Yuan, and Jian Guo. ChartBench: A benchmark for complex visual reasoning in charts. *arXiv preprint arXiv:2312.15915*, 2023.

[32] Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, et al. CharXiv: Charting gaps in realistic chart understanding in multimodal LLMs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[33] Renqiu Xia, Bo Zhang, Hancheng Ye, Xiangchao Yan, Qi Liu, Hongbin Zhou, Zijun Chen, Min Dou, Botian Shi, Junchi Yan, and Yu Qiao. ChartX & ChartVLM: A versatile benchmark and foundation model for complicated chart reasoning. *arXiv preprint arXiv:2402.12185*, 2024. The ChartVLM model is described in the same paper.

[34] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[35] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*, 2024.

[36] Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[37] Jinyue Chen, Lingyu Kong, Haoran Wei, Chenglong Liu, Zheng Ge, Liang Zhao, Jianjian Sun, Chunrui Han, and Xiangyu Zhang. OneChart: Purify the chart structural extraction via one auxiliary token. *arXiv preprint arXiv:2404.09987*, 2024.

[38] Fangyu Liu, Francesco Piccinno, Syrine Krichene, Chenxi Pang, Kenton Lee, Mandar Joshi, Yasemin Altun, Nigel Collier, and Julian Martin Eisenschlos. MatCha: Enhancing visual language pretraining with math reasoning and chart derendering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023.

[39] Ahmed Masry, Parsa Kavehzadeh, Xuan Long Do, Enamul Hoque, and Shafiq Joty. UniChart: A universal vision-language pretrained model for chart comprehension and reasoning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

[40] Fanqing Meng, Wenqi Shao, Quanfeng Lu, Peng Gao, Kaipeng Zhang, Yu Qiao, and Ping Luo. ChartAssistant: A universal chart multimodal language model via chart-to-table pre-training and multitask instruction tuning. In *Findings of the Association for Computational Linguistics: ACL 2024*, 2024.

[41] Yucheng Han, Chi Zhang, Xin Chen, Xu Yang, Zhibin Wang, Gang Yu, Bin Fu, and Hanwang Zhang. ChartLlama: A multimodal LLM for chart understanding and generation. *arXiv preprint arXiv:2311.16483*, 2023.

[42] Renqiu Xia, Haoyang Peng, Hancheng Ye, Mingsheng Li, Xiangchao Yan, Peng Ye, Botian Shi, Yu Qiao, Junchi Yan, and Bo Zhang. StructChart: On the schema, metric, and augmentation for visual chart understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(7):8044–8059, 2026. doi: 10.1109/TPAMI.2026.3669664.

[43] Ahmed Masry, Megh Thakkar, Aayush Bajaj, Aaryaman Kartha, Enamul Hoque, and Shafiq Joty. ChartGemma: Visual instruction-tuning for chart reasoning in the wild. *arXiv preprint arXiv:2407.04172*, 2024.

[44] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision (ECCV)*, 2018.

[45] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R. Manmatha, Mu Li, and Alexander Smola. ResNeSt: Split-attention networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2022.

[46] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. UNet++: Redesigning skip connections to exploit multiscale features in image segmentation. *IEEE Transactions on Medical Imaging*, 2020.

[47] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

[48] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[49] Junyu Luo, Zekun Li, Jinpeng Wang, and Chin-Yew Lin. ChartOCR: Data extraction from charts images via a deep hybrid framework. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021.

[50] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. LayoutLMv3: Pre-training for document AI with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia (ACM MM)*, 2022.

[51] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1290–1299, 2022.

[52] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision (ECCV)*, 2020.

[53] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M. Alvarez, and Ping Luo. SegFormer: Simple and efficient design for semantic segmentation with transformers. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[54] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment Anything. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[55] Jitesh Jain, Jiachen Li, Mang Tik Chiu, Ali Hassani, Nikita Orlov, and Humphrey Shi. OneFormer: One transformer to rule universal image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[56] Ankit Rohatgi. WebPlotDigitizer: Version 4.6. `https://automeris.io/WebPlotDigitizer`, 2022.

[57] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, and Kristin A. Persson. Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1):011002, 2013. doi: 10.1063/1.4812323.

[58] Alexander Dunn, Qi Wang, Alex Ganose, Daniel Dopp, and Anubhav Jain. Benchmarking materials property prediction methods: the Matbench test set and Automatminer reference algorithm. *npj Computational Materials*, 6:138, 2020. doi: 10.1038/s41524-020-00406-3.

[59] Andres M. Bran, Sam Cox, Oliver Schilter, Carlo Baldassari, Andrew D. White, and Philippe Schwaller. Augmenting large language models with chemistry tools. *Nature Machine Intelligence*, 2024. doi: 10.1038/s42256-024-00832-8.

[60] Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624:570–578, 2023. doi: 10.1038/s41586-023-06792-0.

[61] Glenn Jocher and Jing Qiu. Ultralytics YOLO11, 2024. URL `https://github.com/ultralytics/ultralytics`. Version 11.0.0.

[62] Stuart P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[63] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

A Materials-Science Chart Families: A Glossary for Non-Specialists

The body of this paper uses the standard materials-science abbreviations without expansion to keep the prose compact. We summarise here what each family measures and what its x/y axes mean, since these conventions ground the reasoning tasks of §7 and the per-family breakdown of Appendix G.

- **XRD** (X-ray diffraction): intensity (a.u.) vs. diffraction angle 2θ (degrees). Used to identify crystalline phases and refine lattice parameters; peaks at characteristic 2θ values are indexed by Miller-index labels such as "(020)", "(111)".
- **XPS** (X-ray photoelectron spectroscopy): intensity (a.u.) vs. binding energy (eV, conventionally plotted with the x-axis reversed: high $\rightarrow$ low). Used for surface elemental composition and chemical-state assignment; peak labels such as "Mn $2p_{3/2}$" name the core-level orbital being probed.
- **Raman** (Raman scattering): intensity (a.u.) vs. Raman shift (cm^{-1}). Used for vibrational modes; the $\mathrm{I_D}/\mathrm{I_G}$ peak-intensity ratio is the standard disorder/graphitisation proxy in carbon materials.
- **XAFS / EXAFS** (X-ray absorption fine structure): absorption $\mu(E)$ vs. photon energy (eV). Used for oxidation-state assignment (XANES region near the absorption edge) and local atomic coordination (EXAFS oscillations above the edge).
- **EIS** (electrochemical impedance spectroscopy): the most common rendering is a Nyquist plot of $-Z''$ vs. Z' (impedance, Ω); sometimes Bode plots of $|Z|$ and phase vs. frequency. Used for charge-transfer resistance and double-layer capacitance in batteries and supercapacitors. The high-frequency Z' intercept that the B-EIS task of §7 extracts is the series (electrolyte) resistance.
- **GCD** (galvanostatic charge–discharge): cell voltage (V) vs. specific capacity (mAh g^{-1}). The plateau regions and the endpoint capacity are the headline performance numbers for battery electrodes.
- **CV** (cyclic voltammetry): current (A or A g^{-1}) vs. potential (V). The peak positions and shapes diagnose redox reactions; the enclosed area scales with capacitance.
- **C-Rate / Rate-capability**: discharge capacity (mAh g^{-1}) vs. cycle index, with each segment recorded at a different C-rate (e.g. 0.1 C, 0.5 C, 1 C). Used to test rate performance.
- **DOS** (density of states): density of electronic states vs. energy ($E - E_F$, eV). Used for band-structure analysis from DFT calculations.
- **IR** (infrared spectroscopy): transmittance or absorbance vs. wavenumber (cm^{-1}). Used for functional-group identification via vibrational fingerprints.

Three publication conventions recur in these families and motivate specific design choices in MatCurvs: **(i) Stacked-intensity offset.** XRD, Raman, XPS and XAFS papers routinely vertically offset successive spectra by a constant intensity for visual clarity; the offset is invisible-yet-measurable in the extracted (x, y) data. This is the basis of the C-Stack task in §7. **(ii) Reversed x-axis (XPS).** XPS plots binding energy high-to-low by convention; pipelines that assume monotonically increasing x silently swap directions. **(iii) Arbitrary intensity units.** Most spectra label the y-axis "Intensity (a.u.)"; absolute intensity carries no physical meaning, only relative peak heights do — which is why Figure F.2 autoscales each method's y-axis independently.

B L0 — Subfigure Routing: Models, Counts, Per-class Accuracy

B.1 Detector training and benchmark

The subfigure detector was trained on $1{,}366$ manually annotated subfigure panels drawn from a stratified sample of the $55{,}763$-article corpus. Each panel was bounded by a single annotator and double-checked by a second. We benchmarked four detector families (Figure B.1): YOLOv11x, Mask-RCNN with ResNet-50, Mask-RCNN with ResNet-101, and a 7B-parameter Qwen-VL baseline operated as a bounding-box predictor. YOLOv11x is the uniform best across all four reported metrics ($\mathrm{mAP}_{0.5}$, $\mathrm{mAP}_{0.5:0.95}$, precision, recall) and is what we use throughout this paper. The corresponding confusion matrix on the held-out test split (Figure B.2) shows $1{,}286$ true positives, 21 false negatives (1.6%) and 59 false positives (4.4%), giving precision 95.6% and recall 98.4% on the binary subfigure-vs-background task.

B.2 Hierarchical chart-type and characterisation classifier

Each cropped subfigure is classified along three axes by a single multimodal-LLM call (qwen-vl-max with a step-by-step chain-of-thought prompt; full prompt released with the code):

- **chart type** (6 classes): line chart, bar chart, scatter plot, heatmap, pie chart, other.
- **first-level data origin** (3 classes): Experimental Data, Computational Data, Schematics / Illustrations.

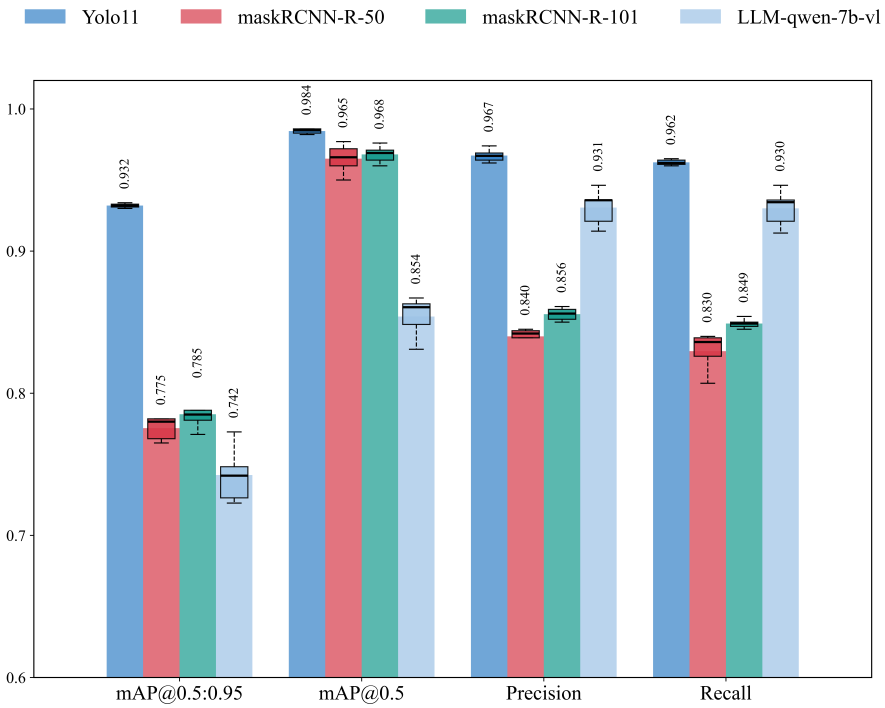

Figure B.1: Subfigure-detector benchmark across four model families. YOLOv11x dominates Mask-RCNN baselines on $\text{mAP}_{0.5:0.95}$ (0.932 vs. 0.775 / 0.785); the 7B Qwen-VL baseline reaches high precision/recall but fails the strict $\text{mAP}_{0.5:0.95}$ test (0.742). Boxes show variance over five training runs.

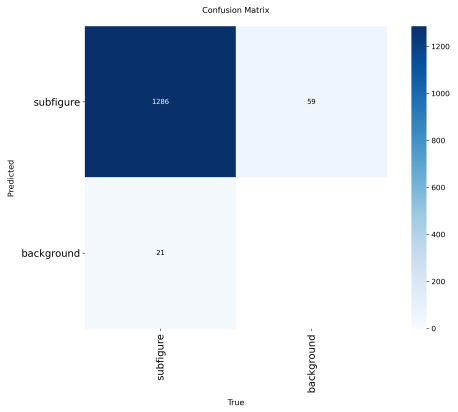

Figure B.2: Confusion matrix of YOLOv11x on the held-out subfigure- detection test split: 1,286 true positives, 21 false negatives, 59 false positives.

- **second-level characterisation** (10 canonical classes): Performance, Microscopy, Structural Characterization, Compositional Analysis, DFT, MD, FEM, Structural diagrams, Process flow diagrams, Conceptual illustrations.

A long tail of LLM-emitted variants (e.g. “Microscopy: SEM”, “DFT: Band Structure”) comprising 0.11% of total entries is merged back to the canonical 10 categories.

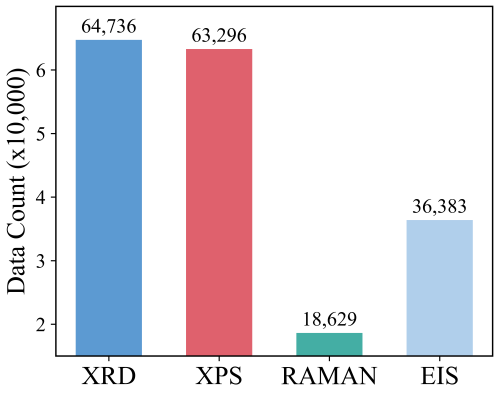

Figure B.3: Characterisation-type counts (XRD / XPS / Raman / EIS).

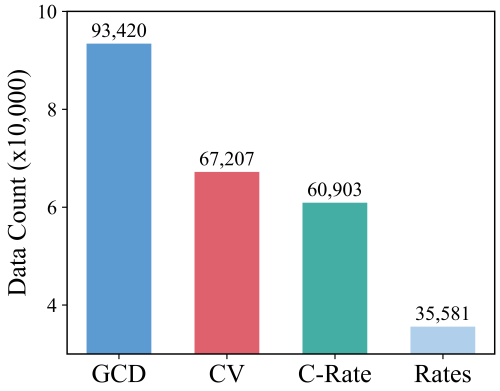

Figure B.4: Performance-type counts (GCD / CV / C-Rate / Rates).

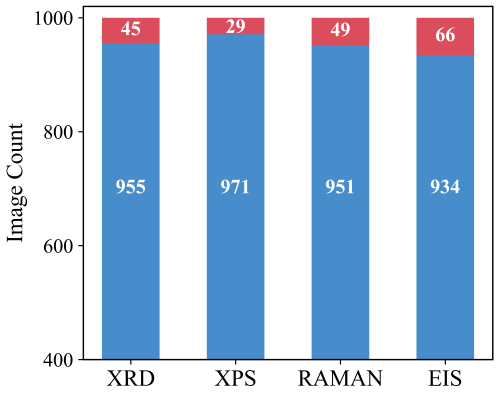

Figure B.5: Per-class classifier accuracy (characterisation), manual audit of n=1,000 per class. Blue = correct, red = failed. XRD 95.5%, XPS 97.1%, Raman 95.1%, EIS 93.4%.

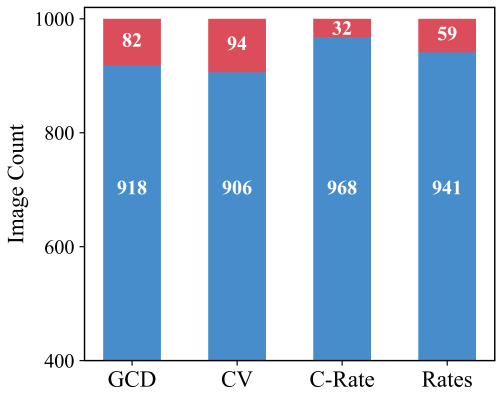

Figure B.6: Per-class classifier accuracy (performance), manual audit of n=1,000 per class. GCD 91.8%, CV 90.6%, C-Rate 96.8%, Rates 94.1%.

B.3 Per-category counts and classifier accuracy

Figure B.3 and Figure B.4 report classified subfigure counts per characterisation type and per electrochemical-performance type, respectively. Figure B.5 and Figure B.6 report per-class accuracy on a manual audit of 1,000 stratified subfigures per class. Mean per-class accuracy is 94.3%; minimum is 90.6% (CV) and maximum is 97.1% (XPS).

C Document-Scale Upstream Pipeline (full tables)

Figure C.1 shows the document-scale workflow in panels (a)–(e), referenced from §5; Table C.1 reports each stage's held-out test performance and Table C.4 the per-category instance counts. Figure C.2 gives the line-extractor cell schematic also referenced in §5.

D Mask2Former Architecture, Training-Data, and Loss Ablations

The default foreground segmenter (q=12, d=3, h=256, MiT-B3 backbone) was selected from a 24-cell sweep over architecture (q, d, h), 40k synthetic charts with or without 64× real-image oversampling, and three loss recipes: **Recipe A** (binary-IoU, our chosen training protocol), **Recipe B**

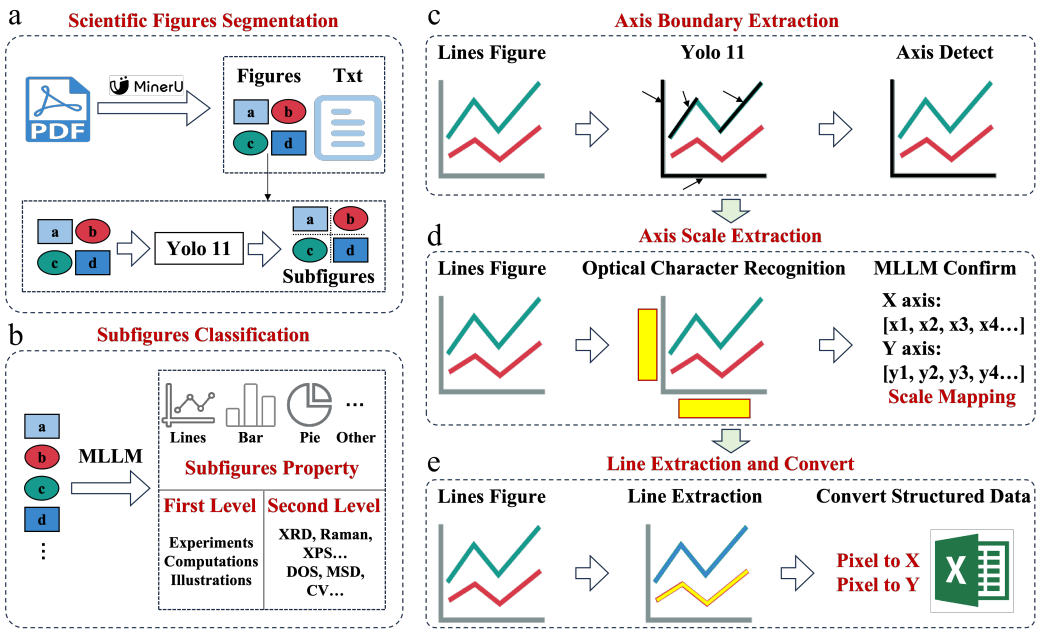

Figure C.1: End-to-end article-to-trajectory workflow. (a) PDF-to-subfigure extraction; (b) subfigure classification and hierarchical property assignment; (c) axis-boundary detection; (d) tick and scale recognition with pixel-to-value mapping; and (e) curve extraction and conversion to data-space sequences.

Algorithm 1 The MatDeplot Stage 3 line-extractor on a single chart image.

Require: image I; thresholds $\theta_{\text{fg}}{=}0.25$, $\tau_{\text{fg}}{=}0.003$; TTA scales $\mathcal{S}{=}\{512, 768, 1024\}$; spatial weight $w_{xy}{=}0.5$.

$B_{\text{axis}}, B_{\text{legend}} \leftarrow \textsc{YoloAxis}(I), \textsc{YoloLegend}(I)$; fit calibration $\mathcal{C}$; build inside-plot mask $\mathcal{V}$.

$F \leftarrow \text{maxprob}\big(\{\text{M2F}_s(I), \text{M2F}_s(\text{flip}\, I)\}_{s\in\mathcal{S}}\big)$.

$P \leftarrow \{F \geq \theta_{\text{fg}}\} \cap \mathcal{V}$; $\rho \leftarrow |P|/(H \cdot W)$.

$k \leftarrow \textsc{KSource}(I)$ ▷ cached Gemini one-shot, or silhouette in $\{2, \ldots, 6\}$.

if $\rho < \tau_{\text{fg}}$ **then**

  $\hat{S} \leftarrow \textsc{KMeans}_k\big(\phi[\mathcal{V} \setminus \textsc{Otsu}(I)]\big)$; `used_fallback`←True.

else

  $\hat{S} \leftarrow \textsc{KMeans}_k\big(\phi[P]\big)$; apply v4 post-processing (morph-close 5, dilate 3, min-area 200, ΔE-gated dilation); `used_fallback`←False.

end if

$\{\pi_i\} \leftarrow \textsc{SkeletoniseAndOrder}(\hat{S})$; calibrate via $\mathcal{C}$.

return $\{\pi_i\}$ with per-curve confidence and `used_fallback` flag.

(dice + focal, real-augment ablation), and **Recipe C** (multi-task instance + trajectory + style + endpoint, plus BCE-only controls). All variants share an identical optimiser configuration: AdamW, 100 epochs, batch size 8, image size 640, lr=$1{\times}10^{-4}$ on the head and $1{\times}10^{-5}$ on the MiT-B3 backbone, with linear warm-up over 5 epochs and cosine decay; trained on a single H100 80 GB GPU.

Table D.1 reports the best-epoch real validation foreground IoU (the metric the checkpoint scheduler optimises) across 24 configurations and 25 checkpoint rows. Table D.2 additionally re-uses four checkpoints under the full end-to-end pipeline on the leakage-clean $n{=}90$ hold — the ranking on validation IoU transfers monotonically to the downstream DTW-Δ and CCR ranking. Three points stand out.

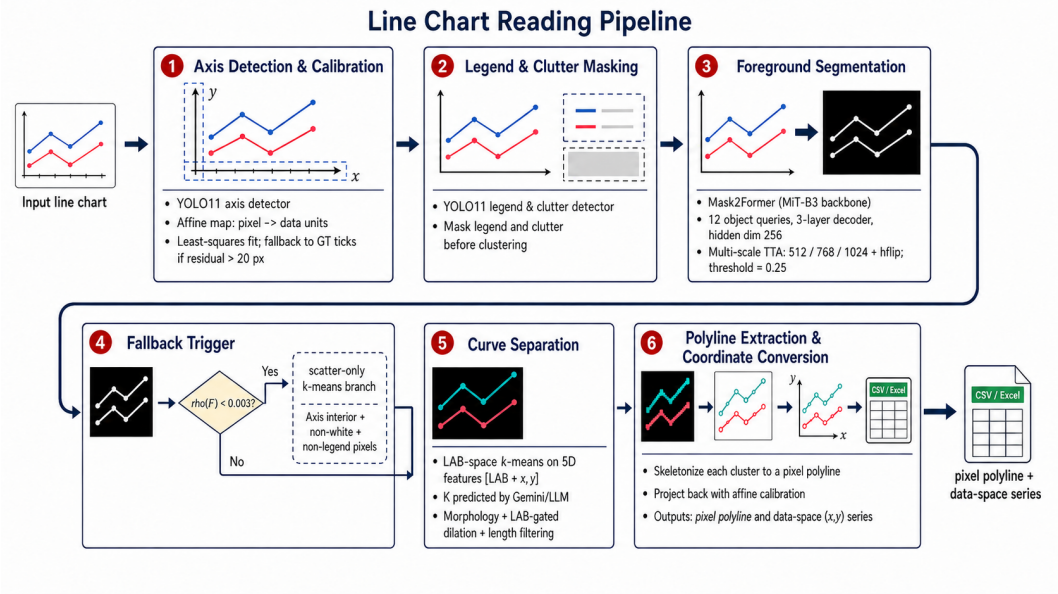

Figure C.2: Line-extractor cell schematic referenced from §5. Six cells: input chart → **(1)** YOLO11 axis detector → **(2)** YOLO11 legend masker → **(3)** Mask2Former foreground segmenter (q=12, d=3, h=256, MiT-B3 backbone, multi-scale + flip TTA, threshold θ_{fg}=0.25) → **(4)** fallback trigger $g(F)$ on the foreground fraction $\rho(F)$, which reroutes to a scatter-only k-means branch when the mask is nearly empty ($\rho(F)<\tau_{\text{fg}}$=0.003) → **(5)** 5D LAB-xy k-means curve separator with v4 post-processing (morph-close 5, dilate 3, min-area 200, ΔE-gated dilation) → **(6)** polyline extraction and coordinate conversion (skeletonise each cluster to a pixel polyline, project back through the affine calibration, attach per-curve confidence and the abstain flag).

Table C.1: Document-scale pipeline stages with held-out test performance. Subfigure-detector results are mean±range of 5 independent training runs; axis and legend detectors are mean±range across the same 5 runs on the line-extraction test sets. Classifier accuracies are stratified by the eight characterization categories on n=1000 images each, hand verified.

Stage	Sub-step	Model	|Test|	Headline metric	Source
1	1.1 PDF parsing	MinerU	55,763 docs	—	external
	1.2 Subfigure splitting	**YOLO11x** (this work)	—	mAP_{50}=98.4%, $\text{mAP}_{50:95}$=93.2%	Tab. C.2
	(α comparison)	Mask R-CNN R-50	—	mAP_{50}=96.5%, $\text{mAP}_{50:95}$=76.9%	Tab. C.2
	(α comparison)	Mask R-CNN R-101	—	mAP_{50}=97.4%, $\text{mAP}_{50:95}$=78.6%	Tab. C.2
2	Hierarchical classifier	qwen-vl-max	8×1000	mean per-class acc=94.3%	Tab. C.3
3	3.1 Axis bbox detection	YOLO11 (this work)	108 instances	mAP_{50}=99.4%, $\text{mAP}_{50:95}$=99.3%	—
	3.2 Legend / clutter bbox	YOLO11 (this work)	594 instances	mAP_{50}=88.8%, $\text{mAP}_{50:95}$=75.4%	—
	3.3 Foreground segmentation	Mask2Former (this work)	—	—	§5
	3.4 Per-curve separation	5D LAB-xy k-means (§5)	90 manual	$\text{LR}_{0.5}$=45.8%	Tab. 3

- *Architecture knobs are sub-dominant.* Across the $\{q\in\{8,10,12\}, d\in\{3,4\}, h=256\}$ subgrid under Recipe A, the spread of best-epoch real-IoU is ±0.005 around the chosen point; no (q,d) gain exceeds the validation noise floor.
- *Real-image oversampling improves validation IoU.* Moving from synthetic-only to 64× real-image oversampling (synth-only vs. real×64 rows of Table D.1), shifts real-IoU by $+0.04$–$+0.07$.
- *Multi-task loss design is mostly neutral.* Recipe C (multi-task) sits within ±0.03 of the simpler Recipe A binary-IoU; a BCE-only control is -0.03 behind the IoU loss; removing the trajectory term (*no-traj*) or shrinking the dataset to the Recipe-B size cap (*B-fair*, *B-size*) loses 0.01–0.05 IoU.

Table C.2: Subfigure detector head-to-head on the same 5-run train/val split. YOLO11x beats both Mask R-CNN backbones on every metric and every individual run; the $+15$ pp $\text{mAP}_{50:95}$ gap is the dominant lever for the downstream classifier.

Detector	precision	recall	mAP_{50}	$\text{mAP}_{50:95}$
YOLO11x (ours)	**0.967**$\pm$0.005	**0.962**$\pm$0.003	**0.984**$\pm$0.002	**0.932**$\pm$0.002
Mask R-CNN R-50	0.838$\pm$0.011	0.822$\pm$0.013	0.965$\pm$0.011	0.769$\pm$0.020
Mask R-CNN R-101	0.850$\pm$0.014	0.840$\pm$0.020	0.974$\pm$0.004	0.786$\pm$0.021

Table C.3: Multimodal-LLM classifier accuracy per characterization category on n=1000 hand-verified subfigures per category. **Failed** counts mis-routed subfigures; accuracy is $1-\text{failed}/1000$.

	Structural characterisation					Electrochemical / kinetic			
	XRD	XPS	Raman	EIS	mean	GCD	CV	C-Rate	Rates
failed (out of 1000)	45	29	49	66	47.3	82	94	32	59
accuracy	95.5%	97.1%	95.1%	93.4%	95.3%	91.8%	90.6%	96.8%	94.1%

The combination "Recipe A architecture + 40k synthetic + 182 real charts, binary-IoU loss" (the released `q12_d3_h256` checkpoint) is therefore preferred not because it dominates every alternative, but because it sits inside the flat plateau on validation IoU while having the simplest loss recipe and the cleanest end-to-end downstream profile (Table D.2).

E Per-Stratum Tables and Auxiliary Sweeps

Table E.2 reports the oracle-FG ceiling (M2–M6) and the UNet++/ResNeSt M4 row on the legacy 36 set. Table E.1 reports the shape-mode and pixel-mode results side-by-side on the legacy 36 set and the n=90 hold (the two modes share the entire backbone and differ only in the Cell 5 post-processing block; §5). Table E.3 reports the per-stratum split for our MATDEPLOT (shape mode) and MATDEPLOT-NOFG rows on the legacy 36 set; Table E.4 is the matching per-stratum breakdown on the leakage-free n=90 leakage-clean subset. Table E.5 is the live-API VLM head-to-head on that same subset. Table E.6 is the fallback threshold sweep, Table E.7 the perturbation table.

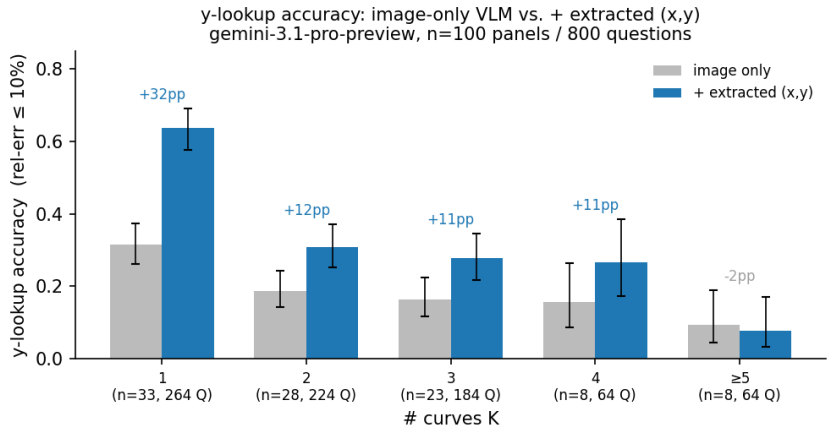

Figure E.1: Per-K visualisation of Table E.8: y-lookup accuracy of Gemini-3.1-pro on the L1 label set (n=100 panels, 800 questions, $\pm 10\%$ rel-err tolerance) under condition A (image only) vs. B (image plus our extracted (x, y) data). Error bars are Wilson 95% CIs on the binomial accuracy at each bin. The K=5 (n=5 panels, 40 Q) and $K \geq 6$ (n=3 panels, 24 Q) tail bins of Table E.8 are merged into a single $K \geq 5$ bin (n=8 panels, 64 Q) here, since the per-K Wilson intervals on those tail rows ($\pm$15–19 pp) are too wide to read individually. The K=1–4 bins all show significant gain from extraction ($+$11–32 pp); the $K \geq 5$ bin's CIs overlap, indicating that on charts with five or more curves the extraction quality at this scale is no longer sufficient to help VLM y-lookup.

Table C.4: Per-category line-chart instance counts produced by the end-to-end document pipeline on the 55,763-paper corpus, restricted to the eight named characterisation categories. The sub-total (440,155) does not exhaust the 657,428 line plots reported in §4.1: ∼217k additional panels fall into DOS, IR, EXAFS, and other minor characterisation tags whose per-class counts are not separately broken out here.

Structural characterisation	count	Electrochemical / kinetic	count
XRD	64,736	GCD	93,420
XPS	63,296	CV	67,207
Raman	18,629	C-Rate	60,903
EIS	36,383	Rate-capability	35,581
sub-total	183,044	sub-total	257,111
All categories			**440,155**

Table D.1: Architecture / data / loss sweep on the held-out real validation split (24 configurations, with a second seed for the main multi-task configuration; 25 checkpoint rows, best epoch per row). REAL_IOU is the best-epoch real-image foreground IoU (the metric the checkpoint scheduler optimises). "Raw" / "mask" distinguishes two label-generation strategies on the 40k synthetic slice. REAL×K = real images repeated $K\times$ during training. Recipes: A = binary-IoU (default), B = dice + focal, C = multi-task (instance + trajectory + style + endpoint).

Recipe	Variant	q	d	h	slice	loss / notes	real_iou
A	q8_d3_h256 (raw)	8	3	256	40k raw	binary-IoU	0.5498
	q10_d3_h256 (raw)	10	3	256	40k raw	binary-IoU	0.5531
	q10_d4_h256 (raw)	10	4	256	40k raw	binary-IoU	0.5531
	q12_d3_h256 (default)	12	3	256	40k raw	**binary-IoU**	**0.5557**
	q8_d3_h256 (mask)	8	3	256	40k mask	binary-IoU	0.5552
	q10_d3_h256 (mask)	10	3	256	40k mask	binary-IoU	0.5371
	q10_d4_h256 (mask)	10	4	256	40k mask	binary-IoU	0.5481
	q12_d3_h256 (mask)	12	3	256	40k mask	binary-IoU	0.5456
B	q8_d3_h256 (synth-only)	8	3	256	40k, synth-only	dice+focal	0.4864
	q8_d3_h256 (real×64)	8	3	256	40k, real×64	dice+focal	0.5279
	q10_d3_h256 (synth-only)	10	3	256	40k, synth-only	dice+focal	0.4837
	q10_d3_h256 (real×64)	10	3	256	40k, real×64	dice+focal	0.5371
	q10_d4_h256 (synth-only)	10	4	256	40k, synth-only	dice+focal	0.4858
	q10_d4_h256 (real×64)	10	4	256	40k, real×64	dice+focal	0.5367
	q12_d3_h256 (synth-only)	12	3	256	40k, synth-only	dice+focal	0.4828
	q12_d3_h256 (semantic-only)	12	3	256	40k, synth-only	dice (semantic only)	0.4844
	q12_d3_h256 (real×64)	12	3	256	40k, real×64	dice+focal	0.5484
C	main (seed 0)	12	3	256	40k, real×64	multi-task	0.5322
	main (seed 1)	12	3	256	40k, real×64	multi-task	0.5352
	queries=10	10	3	256	40k	multi-task	0.5093
	queries=30	30	3	256	40k	multi-task	0.5117
	no-trajectory loss	12	3	256	40k	multi-task, − trajectory	0.5061
	BCE-only control	12	3	256	40k	BCE-only	0.5269
	B-fair (size-matched)	12	3	256	40k (size-matched to B)	multi-task	0.5062
	B-size (size cap)	12	3	256	40k (B-size cap)	multi-task	0.5243

F Qualitative Comparison Across Extractors

We present two complementary views of how MatDeplot compares to the alternative extractors. Figure F.1 stays in *pixel space*: it overlays MatDeplot's per-line cluster masks on the source chart for four L1 panels (one per stratum), showing that the cluster colours sit *on top of* the GT polylines without drift — the visual evidence behind the $\mathrm{LR}_{0.5}$=45.8% headline of Table 3. Figure F.2 then moves to *real-coordinate space*: on 10 L2 panels stratified by $K\in\{1,\ldots,6\}$ it plots Ours' L2-curated (x, y) alongside what each VLM and DePlot extracted under the same uniform prompt —

Table D.2: End-to-end FG-checkpoint swap on the leakage-controlled leakage-clean n=90 subset, holding YOLO axis, YOLO legend, LAB-k-means separator, and skeleton-polyline extractor fixed (no morph post-processing, no flip TTA, threshold 0.35, scales $\{640, 1024\}$ — a controlled FG-only ablation; not the shape-mode or pixel-mode recipe of Tables 3 and E.1). The DTW and CCR ranking is monotone with the real-IoU ranking of Table D.1.

Checkpoint	DTW↓	CCR↑	Inst-F1↑	PQ↑	fb
q12_d3_h256, 40k synth + 182 real (default)	**19.92**	**0.898**	0.650	**0.204**	0/90
q10_d3_h256, 20k synth + 182 real	23.78	0.876	0.672	0.191	1/90
q15_d3_h256, 20k synth + 182 real	23.27	0.877	0.690	0.178	1/90
q10_d6_h256, 30k synth + 182 real	22.80	0.880	**0.691**	0.202	0/90

Table E.1: Shape-mode vs. pixel-mode of MatDeplot's Cell 5 (the two operating points share Cells 1–4 and Cell 6; only the morph-close + min-area + LAB-ΔE dilation block differs). On the legacy 36 set the shape mode wins shape-regime metrics; the pixel mode wins pixel-regime metrics on the n=90 hold.

	DTW↓	CCR↑	Inst-F1↑	Curve-PQ↑	$\mathrm{LR}_{0.5}$ ↑	$\mathrm{LR}_{0.8}$ ↑
the legacy 36*-image set*						
shape mode	**14.7**	**0.964**	0.520	**0.178**	—	—
pixel mode	23.2	0.951	**0.572**	0.031	—	—
leakage-clean n=90 *hold (cached Gemini-K)*						
shape mode (gemK)	21.11	0.878	0.929	**0.229**	—	—
pixel mode (gemK)	**19.93**	**0.917**	**0.942**	0.126	**45.8%**	**0.5%**

LR columns are not reported for the legacy 36 set because the unified evaluator on that split records DTW/CCR/Inst-F1/Curve-PQ only; the LR sweep is computed against the dedicated 5-pixel GT rasteriser used on the n=204 side, and we leave that cell blank rather than back-fill across rasterisers. The shape-mode LR cells on n=90 are also blank: shape mode's mask is thinner than the 5-pixel GT rasterisation, so a strict-IoU sweep would be floor-bound; the matching reference is the upper bound $\mathrm{LR}_\theta \le \mathrm{PQ}/\theta$ ($0.229/0.5$=45.8%). The pixel mode is the LR-tuned recipe.

showing whether the methods placed their curves at the right physical x interval and whether the recovered shape matches.

Figure F.2 compares MatDeplot, the four hosted VLMs (GPT-5.4 [22], Gemini-3.1-pro [24], Qwen-3.5-plus [25], Claude Opus 4.6 [23]) and DePlot [6] on 10 panels drawn from the L2-curated corpus (ğ4.3), stratified by GT curve count $K \in \{1, 2, 3, 4, 5, 6\}$. Each VLM was queried with the *identical* `DEFAULT_PROMPT` used for Table E.5 — no per-image prompt tuning, so shape/position differences are attributable to the model rather than to prompt drift. *Ours* (column 2) is rendered directly from the screened (x, y) table that ships with each L2 panel; the VLM and DePlot columns plot the methods' raw outputs in the same physical x-range as Ours so the reader can immediately see whether each method placed its curves at the right 2θ / binding-energy / time interval. Per-method y is autoscaled within its panel because the four VLMs use mutually inconsistent y normalisations (some emit raw intensity, others rescale to $[0, 1]$); forcing them all into Ours' y-range hides their shapes entirely. Three robust patterns:

- K **recovery degrades sharply with** K**.** For K=1 every method (except DePlot, which sometimes empty-predicts on scatter-like XPS surveys) returns one trail; for $K \ge 5$ only Claude Opus 4.6 and GPT-5.4 reliably return K_{pred}=K, and Gemini frequently truncates the JSON mid-curve under its 4k-token cap.
- **Shape vs. position.** When a VLM does emit a polyline with K_{pred}=K, the gross shape (peak count, monotonic trend) is usually right; the x-axis placement of features can drift by tens of percent (visible as horizontal offsets between the VLM panel and Ours), confirming the $\mathrm{LR}_\tau \approx 0$ headline of Table 3.
- **DePlot collapses on scientific spectra.** It returns ≤ 2 short series on most panels (`n_predicted_trails` reported below each panel), in line with the existing literature's finding that chart-to-table models trained on bar/pie/short-line charts truncate after a handful of points.

One implementation note for reproducibility: Gemini responses that overflowed the 4k-token `max_tokens` budget mid-JSON are salvaged with a regex-based partial-parse pass that walks

Table E.2: Full oracle-FG ceiling (M2–M6) plus the M4 UNet++/ResNeSt row on the legacy 36-image set. M_i is the i-th separator cell of the MatDeplot oracle-FG harness.

Method	DTW-$\Delta \downarrow$	CCR@0.1$\uparrow$	Inst-F1$\uparrow$	Family
M2 (oracle FG)	113.8	0.865	0.437	oracle ceiling
M3 (oracle FG)	102.3	0.963	0.434	oracle ceiling
M4 (oracle FG)	69.0	0.982	0.138	oracle ceiling
M5 (oracle FG)	127.1	0.782	0.437	oracle ceiling
M6 (oracle FG)	101.1	0.942	0.265	oracle ceiling
M4 (UNet++/ResNeSt)	93.2	0.916	0.279	prior local-FG

Table E.3: Per-stratum performance of our two pipeline variants. Per-image means; the n=1 L3 row is shown for completeness only.

	MatDeplot				MatDeplot-NoFG			
Stratum	DTW	CCR	F1	PQ	DTW	CCR	F1	PQ
ALL (n=36)	**15.9**	**0.932**	0.601	0.233	25.3	0.852	0.471	0.076
L1 single (n=17)	8.2	0.988	0.471	0.237	18.8	0.920	0.312	0.054
L2 multi-disjoint (n=6)	22.4	0.937	0.658	0.319	34.5	0.775	0.571	0.093
L3 multi-overlap (n=1)	2.1	1.000	0.800	0.619	14.5	0.853	0.667	0.274
L4 same-colour (n=6)	23.9	0.885	0.744	0.166	28.8	0.834	0.688	0.142
L5 dense scatter (n=6)	25.4	0.805	0.571	0.140	32.7	0.770	0.523	0.051

`"line_id"`-delimited segments and pulls every well-formed $[x, y]$ pair. The 10-panel manifest, all per-(method, panel) raw responses, and the figure-build script are released alongside the dataset.

Throughput comparison against WebPlotDigitizer. WebPlotDigitizer [56] is the canonical manual-annotation tool that materials-science authors and literature-mining workflows reach for when they want (x, y) data out of a published chart. The user calibrates the two axes by clicking three reference points each, then either marks every data sample on every curve by hand or invokes an automatic-tracing wand that requires per-image colour tuning. In our own use, calibration plus per-curve tracing on a typical multi-curve scientific panel costs 2–5 minutes per chart end-to-end (single-curve panels are closer to 1 minute, dense $K \geq 5$ stacks closer to 5). MatDeplot extracts the same artifact in 1.5 s per image on a single H100 (Table I.1), an 80–200$\times$ wall-clock speed-up at the cost of the IoU–1 residual extraction error quantified in §6: $\mathrm{LR}_{0.5}$=45.8% rather than the $\sim$ 100% that careful manual tracing in WebPlotDigitizer would produce. WebPlotDigitizer therefore remains the right tool for the single-panel case where IoU–1 correctness matters and the user is willing to spend a few minutes per chart; MatDeplot is the right tool for the article-scale and corpus-scale workflows (55,763 articles, 1.37 M classified panels, 3,817 L2 panels) for which manual tracing is operationally infeasible.

G Per-Chart-Type Faithfulness Breakdown

Table G.3 stratifies LR_τ by the seven materials-science chart types we observe in the MatCurvs-204 corpus. Chart-type labels come from the provenance-aware multimodal-LLM classifier of Stage 2 (Subfigure Classification, Figure C.1b); 174/204 images are typed, the remaining 30 are flagged "unknown". The single most important takeaway is that the *distribution of* LR_τ *across chart types is heavily uneven for our pipeline* (IR and Raman > XRD and XPS > EIS, EXAFS, and DOS), but *flat at zero for every hosted VLM regardless of chart type*. Two implications: *(i)* our pipeline's failure modes are concentrated on rare and visually-different chart types (EIS Nyquist plots: complex-plane arcs, no peaks; EXAFS: wave-like oscillations; DOS: bipolar electronic-structure curves) where the foreground model has ≤ 5 training images; *(ii)* hosted VLMs do *not* get any chart type "for free", even on XRD and Raman families that are common in the training corpora.

Table E.4: MATDEPLOT (pixel mode, v4+TTA: scales $\{512, 768, 1024\}$, flip TTA, θ_{fg}=0.25, morph post-processing, cached Gemini-K) on the leakage-clean n=90 hold. Per-stratum means; *fb* is the fraction of images triggering the fallback.

Stratum	n	DTW↓	CCR↑	Inst-F1↑	PQ↑	fb
ALL	90	**19.93**	0.917	**0.942**	0.126	0.00
L1 single	29	9.89	**0.992**	0.937	**0.195**	0.00
L2 two-line	28	28.94	0.858	0.962	0.060	0.00
L3 few-line (3–4)	28	20.18	0.889	0.923	0.112	0.00
L4 many-line (≥ 5)	5	26.27	0.964	**0.971**	0.170	0.00

Table E.5: VLM head-to-head on the leakage-clean n=90 hold, 2026-Q2 live API calls. Same GT pixel polylines as Table E.4. Latency is wall-clock seconds per image including network round-trip. † Gemini-3.1-pro-preview emits zero trails on every L4 (≥ 5 lines) image; reported DTW averages over the 85 images where Hungarian matching produced any pairs.

Method	DTW↓	CCR↑	Inst-F1↑	PQ↑	latency (s)
MATDEPLOT (pixel, gemK)	**19.93**	0.917	**0.942**	**0.126**	≈ 1.5
GPT-5.4	61.59	0.587	0.915	$< 10^{-3}$	~ 20
Gemini-3.1-pro-preview†	73.42	0.210	0.343	0.006	~ 30
Qwen-3.5-plus	76.40	0.521	0.837	$< 10^{-3}$	~ 107
Claude Opus 4.6	83.35	0.594	0.898	$< 10^{-3}$	~ 69

H Paired Bootstrap: TTA Recipe Ablation

The released MatDeplot pixel-mode recipe (Algorithm 1) folds in three test-time augmentation knobs: a foreground threshold of θ_{fg}=0.25, a multi-scale TTA at $\{512, 768, 1024\}$, and horizontal flip TTA. Table H.1 reports paired-bootstrap CIs on the core headline metrics for two nested comparisons on the leakage-clean n=90 hold (10,000 resamples, image-clustered) so a reader can see how much each knob contributes. **(a) Flip-TTA isolated.** Both arms share the multi-scale set $\{512, 768, 1024\}$ and threshold 0.25; only the flip-TTA toggle differs. **(b) Cumulative recipe.** Compares the released recipe against the prior MatDeplot snapshot (threshold 0.35, scales $\{640, 1024\}$, no flip TTA), so the gap aggregates all three knobs at once. The flip-only comparison shows recall (cov_{gt}) $+2.98$ pp $[+2.71, +3.27]$ and a small precision regression -1.45 pp $[-1.68, -1.22]$, both 95% CIs excluding zero. The cumulative comparison attributes $+9.07$ pp recall to the combined recipe; reading panel (b) as a flip-only effect would triple-count the contribution. We therefore report both so the reader can attribute the gain correctly.

Multiple-comparison note. Each panel reports four CIs over the same paired sample; with no FDR correction the family-wise α is ≈ 0.185. We mark CIs that exclude zero with $\star$ (95% comparison-wise) but readers should treat "simultaneously-significant" claims with appropriate caution.

I Per-Backend Error Modes

Table I.1 reports per-backend rates of two distinct failure modes on the n=90 hold: (a) API/runtime failure (`success`=False), (b) successful API call but empty prediction (`success`=True $\wedge$ `n_predicted_trails`=0), and the median per-call latency. The breakdown clarifies why our LR_τ table reports VLMs at ≈ 0 even when their Inst-F1 is high: GPT-5.4 returns one curve per image but miscalibrates the data-to-pixel map; Gemini-3.1-pro *succeeds* on every call but emits an empty trail list 64.4% of the time; Qwen-3.5-plus is 90% available but spends 128 s per call; Claude Opus 4.6 succeeds on every call and emits a non-empty trail list, but at 69 s per call and with the same ≈ 0 pixel-IoU pattern as the other VLMs.

Table E.6: A2 fallback-trigger threshold sweep on the legacy 36 set (n=36).

τ_{fg}	# trigger fires	DTW-Δ	CCR	Inst.F1
0.001	0	15.88	0.932	0.601
0.003 (default)	0	15.88	0.932	0.601
0.010	0	15.88	0.932	0.601
0.030	19	20.23	0.896	0.521
0.100	36	25.29	0.852	0.471
0.300	36	25.29	0.852	0.471
0.990 (forced)	36	25.29	0.852	0.471

Table E.7: A6 robustness: end-to-end metrics under three perturbations applied to all 36 test images and re-scored by the same evaluator.

Perturbation	DTW-$\Delta\downarrow$	CCR@0.1 $\uparrow$	Inst.F1 $\uparrow$	Curve-PQ $\uparrow$
Clean (reference)	15.88	0.932	0.601	0.233
Rotation $+5^\circ$	35.33	0.831	0.537	0.000
JPEG q=30	17.11	0.924	0.603	0.219
Gaussian σ=5	14.09	0.920	0.568	0.254

J Raman Peak Detection: Detail by Peak Count

Table J.1 reports the headline numbers for the Raman peak-detection pilot referenced from §7 (axis-mapping ablation: v1/v2 use the YOLO axis box for x bounds, v3 uses the cluster mask's left/right extrema), and Table J.2 stratifies the v3 (n=60) results by GT peak count. The headline medians (recall=1.00, MAE=2.58 cm^{-1}, Pearson=0.955) hold uniformly across peak counts, including spectra with 9+ peaks where peak detection is hardest. Median precision rises slightly with peak count, consistent with the intensity-threshold post-filter being more conservative when there are more peaks to choose from.

K Reproducibility Pointers

- **Raw per-image metrics** on the legacy 36 set for the MatDeplot pipeline and its FG-off ablation.
- **Cross-method benchmark snapshot**: 526 rows spanning 16 extractor methods $\times$ up to 36 images.
- **Per-image stratum labels** for every image in the benchmark.
- **Pre-trained Mask2Former checkpoint**, released under CC-BY-SA-4.0.
- **Unified evaluator**: a Python module released with this paper under MIT, exposing the LR_θ, DTW-Δ, CCR, Inst-F1 and Curve-PQ metrics.
- **Commit hash and random seed** documented in the released repository.

L Leakage Audit (MatCurvs-204 vs. FG Training Set)

The leakage audit summarised here is the in-PDF self-contained version of the more detailed audit document released alongside the code, responding to the contamination concern raised in §8 (b).

Audit procedure. The Mask2Former-q12 FG model is trained on a corpus of 182 hand-annotated real charts plus 40k synthetic charts (§5). The MatCurvs-204 evaluation set is hand-annotated on a different roll-up of the same upstream corpus, so file-level overlap with the training set is possible. We run a SHA-256 audit at the bytes-of-image level: for every MatCurvs-204 image, we fingerprint the pixel buffer and check membership against the training root's fingerprint set. 114/204 (55.9%) MatCurvs-204 images have identical SHA-256 to a training image; the remaining 90 images are byte-distinct from every training image.

Table E.8: Per-K y-lookup accuracy under condition A (image only) vs. B (image + ours extraction), Gemini-3.1-pro on n=100 images, 800 questions, $\pm 10\%$ relative-error tolerance.

K	n images	n questions	acc (A)	acc (B)	Δ
1	33	264	0.319	**0.636**	+0.317
2	28	224	0.188	**0.308**	+0.121
3	23	184	0.163	**0.277**	+0.114
4	8	64	0.156	**0.266**	+0.109
5	5	40	0.105	0.075	−0.030
≥ 6	3	24	0.083	0.083	±0.000
ALL	100	800	0.215	**0.388**	**+0.172**

Table G.1: **MatCurvs-Reasoning rigorous evaluation** (n=49 panels, 188 questions; GT from L1 manual real-coordinate annotation, referenced from §7). $\mathrm{medRel} = \mathrm{median}\,|\hat{a} - a^*| / \max(|a^*|, 1)$ across items where the model returned a parseable number; $\mathrm{pass}_{5\%}$ = fraction with $\mathrm{relErr} < 0.05$. Best per method in **bold**.

Method	Model	N	medRel ↓	$\mathrm{pass}_{5\%}$ ↑
M1 (VLM, image only)	qwen-vl-max	188	32.6%	21.3%
	gemini-3.1-pro-preview	188	26.4%	32.4%
	gpt-5.4	188	**15.4%**	**37.2%**
M2 (VLM, image + data)	qwen-vl-max	188	15.9%	28.2%
	gemini-3.1-pro-preview	188	4.7%	46.8%
	gpt-5.4	188	**4.3%**	**53.2%**
M3 (LLM + data)	qwen3-max	188	14.0%	38.3%
	claude-opus-4.6	188	8.5%	44.1%
	gpt-5.4	188	**4.6%**	**52.1%**
M4 (`scipy` oracle)	—	188	6.7%	44.7%

The leakage-clean n=90 hold. We use the 90 byte-disjoint images as the leakage-clean hold and restrict every headline number (§6, Table 3) to that subset. The empirical leakage premium of the pixel-mode recipe is $\mathrm{LR}_{0.5}$=54.7% (contaminated n=204) vs. 45.8% (clean n=90) — a 9 pp drop — and $\mathrm{LR}_{0.8}$=1.1% vs. 0.5% (−0.6 pp). Tighter thresholds compress the gap because tight-IoU is harder to "memorise" through near-duplicates. We report the clean-n=90 numbers as headline throughout.

Audit scope. SHA-256 is exact-match; an image cropped or compressed differently would evade the audit. We additionally checked perceptual-hash nearness (pHash distance ≤ 4 on a 64-bit hash) against the training root and found 0 further hits beyond the SHA-256 set, consistent with the test annotation pipeline being a deduplicated subset of a different file dump. Two confounds remain: (i) the authors of papers in the test set *may* have published other figures used in training (but the FG model trains on chart images not on paper-author bundles, so this transfer is at most distribution-level); (ii) the cached Gemini-3.1-pro k-source (§5, "k-source: hybrid call") was queried on every MatCurvs-204 image including the n=90 hold, an information channel disclosed in §8 (c).

Reproduction. The audit is a deterministic SHA-256 comparison between the training-image manifest and the n=204 benchmark image list, producing the 90-image clean hold and the 114-image contaminated complement. The auditing script and the resulting holdout list are released with the code.

M VLM-as-Judge Benchmark

We use four hosted VLMs as judges of the MatDeplot pixel-mode extraction (v4+TTA + cached Gemini-K, the recipe of Table 3) on every MatCurvs-204 image. The pipeline-extracted polylines are rendered as a high-contrast overlay on the original chart with each curve labelled `#0`..`#K-1`, and the prompt asks for a per-curve PASS/FAIL plus an image-level pass-iff-all aggregate (prompt

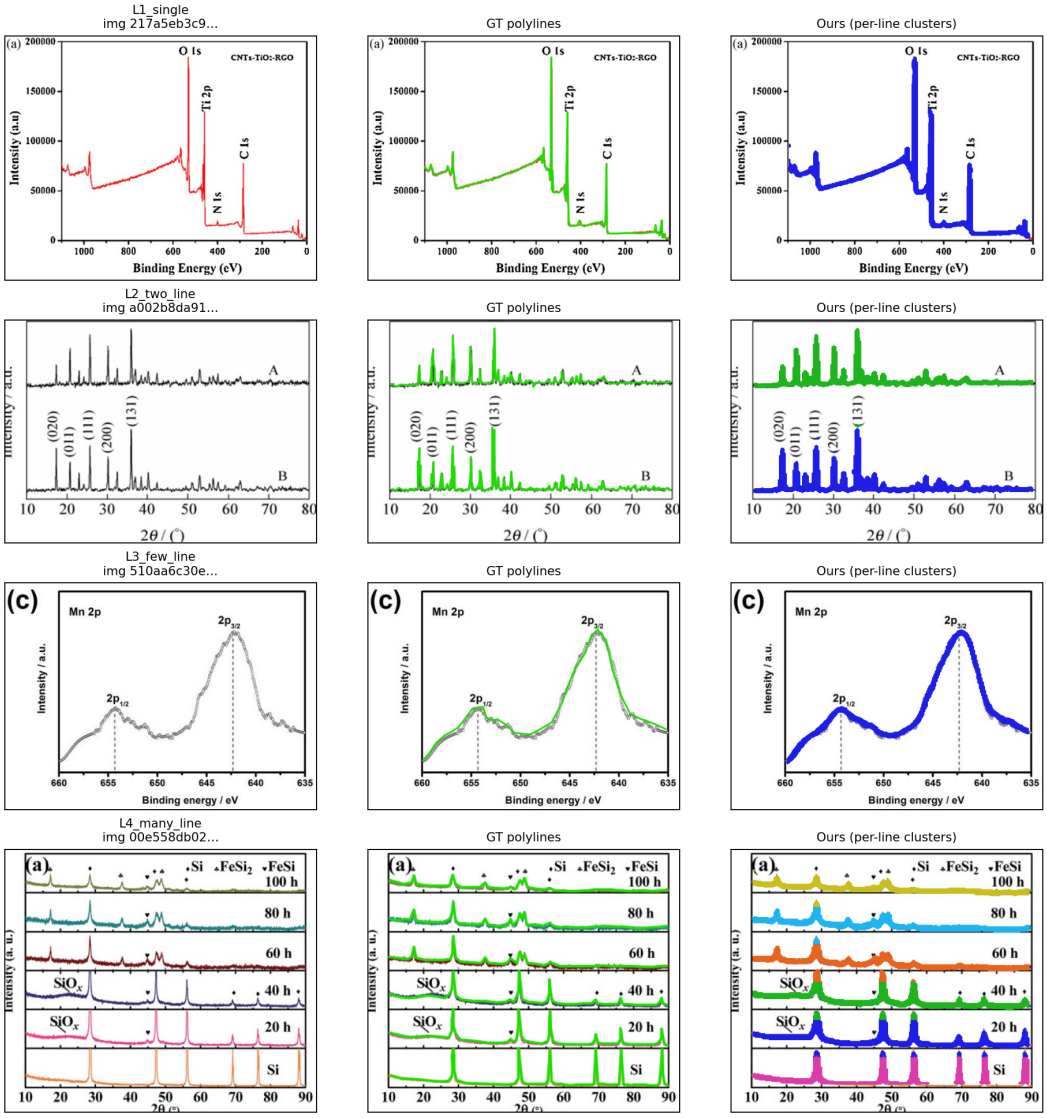

Figure F.1: **Pixel-space reconstruction strip.** Four L1 showcase images (rows: L_1 single XPS, L_2 two-curve XRD, L_3 single-curve Mn 2p, L_4 six-curve XRD stack); columns: original chart, GT polylines (green), MatDeplot's per-line cluster masks (one colour per predicted cluster, drawn directly on the chart pixels). Each panel was picked as the highest-pct_{loc} instance in its stratum subject to $K_{\text{pred}}{=}K_{\text{gt}}$ on the MatDeplot configuration of §6. The cluster colours fall *on* the GT polyline (not adjacent to it) on every row, including the L_4 many-line stack.

verbatim in the released code). All four judges produced valid verdicts: Qwen-VL-Max, GPT-5.4, Gemini-3.1-pro-preview, and Claude Opus 4.6 (the latter on $193/204$ panels). The per-image verdicts are released alongside the dataset.

Per-judge pass rate. Table M.1 reports the image-level pass rate on the full $n{=}204$ set and on the leakage-clean $n{=}90$ hold.

Cross-judge agreement. Table M.2 reports Cohen's κ between every pair of judges over the common image set (judges that errored on a specific image are excluded from that pair only). Values ≥ 0.6 indicate substantial agreement; ≥ 0.8 near-perfect.

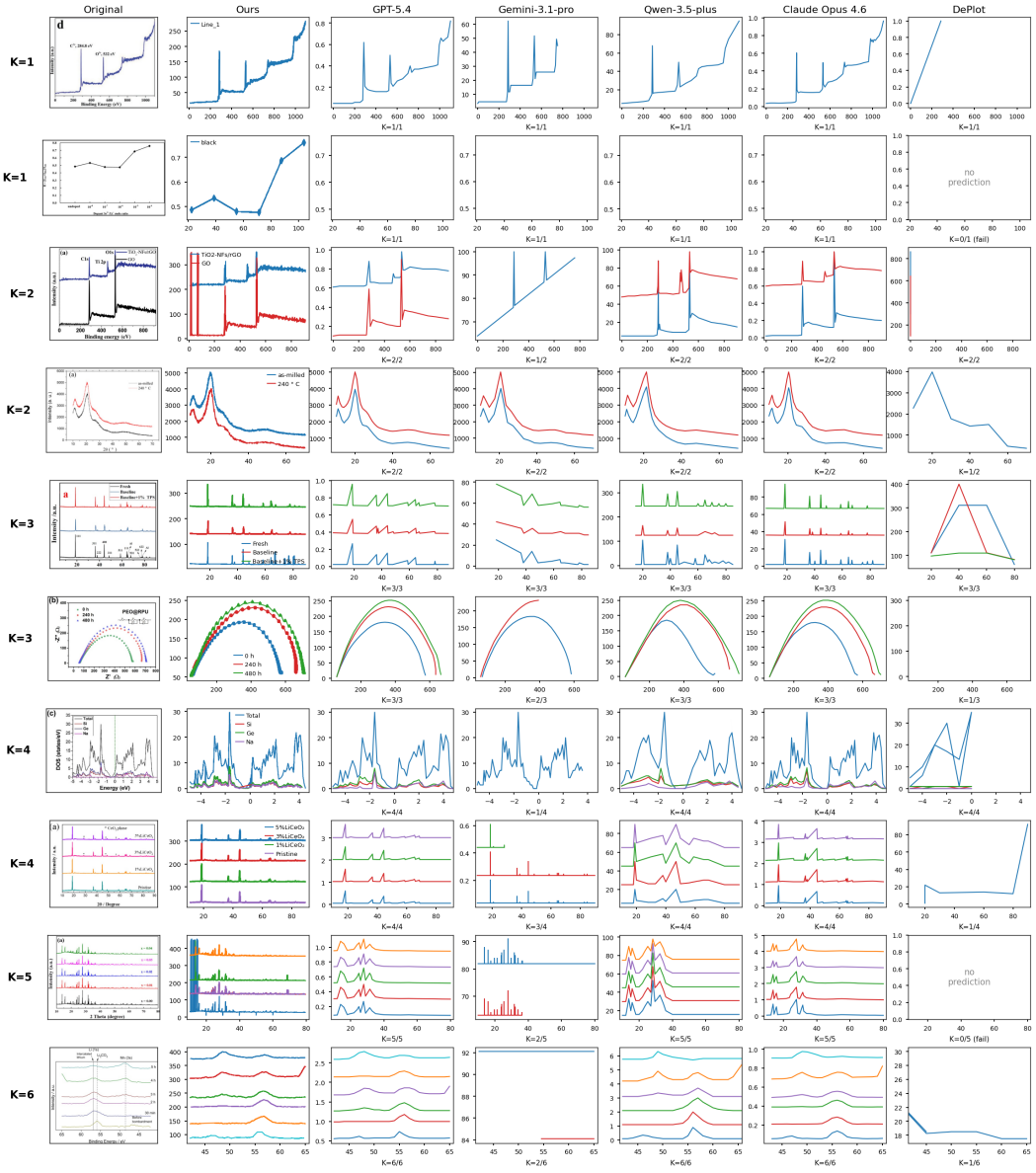

Figure F.2: **Qualitative cross-extractor comparison on** 10 **L2-corpus panels stratified by curve count** $K \in \{1, 2, 3, 4, 5, 6\}$**.** Columns (left to right): original chart image, *Ours* (the L2-curated (x, y) from MatDeplot, plotted as line series, one colour per curve), then the four hosted VLMs and DePlot [6]. Every VLM received the *same* `DEFAULT_PROMPT` (no per-image tuning); each method's x-axis is locked to Ours' physical range so cross-method placement is directly comparable, and per-panel y is autoscaled because the VLMs use inconsistent intensity normalisations. Sub-caption under each non-original panel reports $K_{\text{pred}}/K_{\text{gt}}$ ("fail" when the backend returned an HTTP error).

Table G.2: **Per-task-type breakdown of median relative error** on the n=49 rigorous MR set, referenced from §7. The C-Stack row demonstrates the data-side advantage: image-only median relative error is 23–99%, compared with 2.8–3.5% for the displayed GPT-5.4 and Claude data-using conditions and 62.3% for Qwen M2.

Task	M1 (image)		M2 (image+data)		M3 (data)	
	gpt-5.4	qwen	gpt-5.4	qwen	gpt-5.4	claude-4.6
A1 y *at* x_0	99.7%	99.8%	5.0%	6.9%	6.2%	12.2%
A2 *argmax in* $[a, b]$	0.4%	1.5%	0.4%	2.0%	0.5%	0.6%
A3 *FWHM main peak*	17.6%	32.8%	20.3%	41.6%	20.5%	30.0%
B-Raman *peaks / I_D/I_G*	3.0%	9.5%	2.9%	9.4%	3.3%	9.4%
C-Stack *offset*	**23.1%**	**99.2%**	**3.5%**	**62.3%**	**3.1%**	**2.8%**

Table G.3: LR_τ stratified by chart type on the full n=204 MatCurvs-204 set; "unknown" = images not yet typed by the upstream classifier (treated conservatively, not dropped). VLM rows use the Curve-PQ-derived upper bound of Table 3.

Method	Chart type	n_{img}	n_{lines}	$\mathrm{LR}_{0.5}$	$\mathrm{LR}_{0.6}$	$\mathrm{LR}_{0.7}$	$\mathrm{LR}_{0.8}$	mean IoU
MATDEPLOT	IR	9	15	**86.7%**	**73.3%**	**46.7%**	0.0%	0.637
	Raman	36	74	71.6%	48.7%	16.2%	1.4%	0.558
	XPS	36	67	65.7%	52.2%	25.4%	1.5%	0.555
	XRD	81	180	64.4%	44.4%	16.7%	1.7%	0.537
	EXAFS	5	14	7.1%	0.0%	0.0%	0.0%	0.247
	DOS	3	6	0.0%	0.0%	0.0%	0.0%	0.305
	EIS	4	12	0.0%	0.0%	0.0%	0.0%	0.231
	unknown	30	87	25.3%	16.1%	1.1%	0.0%	0.324
GPT-5.4	all types	203	454	$\leq 0.0\%$	$\leq 0.0\%$	$\leq 0.0\%$	$\leq 0.0\%$	0.002
Gemini-3.1-pro	all types	204	455	$\leq 0.5\%$	$\leq 0.0\%$	$\leq 0.0\%$	$\leq 0.0\%$	0.005
Qwen-3.5-plus	all types	190	421	$\leq 0.0\%$	$\leq 0.0\%$	$\leq 0.0\%$	$\leq 0.0\%$	0.000

Validity: judge flags vs. GT-derived LR. For each judge we compute the mean GT-derived $\mathrm{LR}_{0.5}$ on images the judge *passed* vs. *failed*; a higher $\mathrm{LR}_{0.5}$ on the pass set validates that the judge's flag tracks real reconstruction quality, while $\Delta{=}\mathrm{LR}_{0.5}^{\mathrm{pass}} - \mathrm{LR}_{0.5}^{\mathrm{fail}}$ measures the strength of the correlation.

Consensus rule for Cell 6 abstain. With all four judges in play, Table M.4 sweeps the FAIL-vote threshold from $\geq$ 1-of-4 to $\geq$ 4-of-4 on the n=90 hold. The looser ≥ 1 and ≥ 2 rules abstain on 97.8% and 90.0% of panels respectively (overflagging because individual judges disagree substantially, pairwise κ=0.05–0.27, Table M.2). The $\geq$ 3-of-4 *majority-fail* rule abstains on $64/90$ panels (71.1%) and yields the strongest separation between kept and abstained subsets: $\mathrm{LR}_{0.5}^{\mathrm{kept}}$=0.808 vs. $\mathrm{LR}_{0.5}^{\mathrm{abst.}}$=0.353 ($\Delta$=+0.455). The strictest $\geq$ 4-of-4 *unanimous-fail* rule abstains on only $31/90$ (34.4%) because Claude is structurally more lenient than the other three judges (it agrees with the others on hard failures but not on borderline cases), so the unanimous floor is too rare to be a useful deployment cut. We therefore recommend $\geq$ 3**-of-**4 **majority-fail** as the deployment default; the per-image judge verdicts are released so downstream users can re-curate under any threshold.

Table H.1: Paired bootstrap on Δ, leakage-clean $n{=}90$ hold, 10,000 image-clustered resamples per metric. Panel (a) isolates the flip-TTA contribution; panel (b) is the cumulative gain of the released recipe over the prior snapshot. Comparison-wise 95% CIs; family-wise $\alpha \approx 0.185$ without FDR.

Metric	released	baseline	Δ	95% CI	sig.
(a) Flip-TTA isolated (released vs. released−flip-TTA; same threshold + scales).					
IoU_g	0.6492	0.6415	$+0.0077$	$[+0.0047, +0.0105]$	$\star$
IoU_m	0.4810	0.4729	$+0.0081$	$[+0.0026, +0.0164]$	$\star$
$\mathrm{cov}_{\mathrm{gt}}$	0.8306	0.8008	$+0.0298$	$[+0.0271, +0.0327]$	$\star$
$\mathrm{cov}_{\mathrm{pred}}$	0.7558	0.7703	-0.0145	$[-0.0168, -0.0122]$	$\star$ (worse)
(b) Cumulative recipe (released vs. prior snapshot: threshold $0.35{\to}0.25$*, scales* $\{640, 1024\}{\to}\{512, 768, 1024\}$*, +flip-TTA).*					
IoU_g	0.6492	0.6408	$+0.0084$	$[-0.0010, +0.0178]$	ns
IoU_m	0.4810	0.4768	$+0.0042$	$[-0.0046, +0.0137]$	ns
$\mathrm{cov}_{\mathrm{gt}}$	0.8306	0.7398	$+0.0907$	$[+0.0859, +0.0957]$	$\star$
$\mathrm{cov}_{\mathrm{pred}}$	0.7558	0.8336	-0.0778	$[-0.0868, -0.0694]$	$\star$ (worse)

$\star$ denotes a comparison-wise 95% CI excluding 0. The cumulative panel (b) aggregates three recipe changes (threshold $0.35{\to}0.25$, scale set $\{640, 1024\}{\to}\{512, 768, 1024\}$, flip TTA off→on); panel (a) is the clean flip-only ablation.

Table I.1: Error-mode breakdown on the $n{=}90$ hold. Median latency is the p_{50} across successful calls.

Method	n	success	empty-pred	median p_{50} (s)
GPT-5.4	90	1.000	0.000	19.2
Gemini-3.1-pro	90	1.000	**0.644**	33.1
Qwen-3.5-plus	90	0.900	0.000	128.1
Claude Opus 4.6	90	1.000	0.000	68.6
MatDeplot	90	**1.000**	0.000	**1.5**

Table J.1: Raman peak detection from extracted curves on $n{=}60$ spectra (medians; precision is reported only at v3 because it became meaningful only after the axis-mapping correction). Instrument resolution $\sim 4\,\mathrm{cm}^{-1}$.

Variant	recall (↑)	precision (↑)	MAE cm^{-1} (↓)	Pearson (↑)
v1 (axis_box, n=30)	0.40	0.43	9.64	0.35
v2 (axis_box, n=60)	0.50	—	9.98	0.32
v3 (cluster x, n=60)	**1.00**	**0.71**	**2.58**	**0.96**

Table J.2: Raman peak detection from extracted curves (v3 = cluster-x bounds; n=60), stratified by GT peak count per spectrum.

Peak count bin	n	median recall	median precision	median MAE (cm^{-1})	median Pearson
1–4 peaks	12	1.000	0.667	2.53	0.903
5–8 peaks	14	1.000	0.690	3.41	0.950
9–12 peaks	34	1.000	0.769	2.55	0.958
ALL	60	**1.000**	0.714	**2.58**	**0.955**

Table M.1: Per-judge image-level pass rate on the full $n{=}204$ set and the leakage-clean $n{=}90$ hold. * Claude Opus 4.6 covers $193/204$ ($83/90$ on the hold); the remaining 11 (7) panels are excluded from n_{204}/n_{90}.

Judge	n_{204}	pass_{204}	n_{90}	pass_{90}	median latency (s)
Qwen-VL-Max	204	0.544	90	0.422	4.4
GPT-5.4	204	0.093	90	0.111	5.1
Gemini-3.1-pro-preview	204	0.108	90	0.100	16.1
Claude Opus 4.6*	193	0.420	83	0.386	42.9

Table M.2: Cohen's κ on image-level PASS verdicts (strict intersection n_{common}=193). Lower triangle redundant; diagonal is 1.0 by definition. Pairwise agreement ranges from κ=0.052 (GPT-5.4 vs. Gemini) to κ=0.267 (Qwen-VL-Max vs. Claude Opus 4.6); none reaches the $\kappa \geq 0.6$ "substantial-agreement" threshold, motivating the consensus-rule design in Table M.4.

	Qwen-VL-Max	GPT-5.4	Gemini-3.1-pro	Claude Opus 4.6
Qwen-VL-Max	1.000	0.085	0.111	0.267
GPT-5.4		1.000	0.052	0.058
Gemini-3.1-pro-preview			1.000	0.218
Claude Opus 4.6				1.000

Table M.3: Judge image-level verdicts vs. GT-derived $\text{LR}_{0.5}$ on the leakage-clean n=90 hold. Positive Δ means judge-passed images really are better-reconstructed. * Claude Opus 4.6's $n_{\text{pass}}+n_{\text{fail}}$=83 rather than 90; the remaining $7/90$ hold panels are excluded from this count.

Judge	n^{pass}	$\text{LR}_{0.5}^{\text{pass}}$	$\text{LR}_{0.5}^{\text{fail}}$	Δ
Qwen-VL-Max	38	0.766	0.279	0.487
GPT-5.4	10	0.800	0.445	0.355
Gemini-3.1-pro-preview	9	0.889	0.440	0.449
Claude Opus 4.6*	32	0.641	0.427	0.214

Table M.4: Cell-6 consensus-abstain sweep on the n=90 hold across all four judges: abstain rate vs. GT-derived $\text{LR}_{0.5}$ on the kept and abstained subsets, for thresholds of $\geq 1, 2, 3, 4$ judges flagging FAIL. We adopt the $\geq$ 3-of-4 majority-fail rule (highest Δ) as the deployment default.

$\geq$ FAIL votes	abstain rate	$\text{LR}_{0.5}^{\text{kept}}$	$\text{LR}_{0.5}^{\text{abst.}}$	Δ
1	0.978	0.500	0.484	0.016
2	0.900	0.889	0.440	0.449
3 (default)	**0.711**	**0.808**	**0.353**	**0.455**
4 (unanimous)	0.344	0.626	0.215	0.411

N Absolute Coordinate Quality

On the clean-90 hold (203 curves), a sampled GT location is supported when its matched skeleton has a point within ± 1 image column. Vertical error is measured separately; unsupported locations receive a $100\%H$ penalty in failure-inclusive summaries, where H is full image height.

Table N.1: Coordinate quality and full-mask line recall on the clean-90 hold. Brackets give panel-bootstrap 95% intervals.

Metric	Result
Point support within ± 1 column	$196{,}461/229{,}950 = 85.4\%$ $[80.9\%, 89.8\%]$
Supported median vertical error	2 px ($0.47\%H$)
Failure-inclusive per-curve median / mean	$4.22\%H$ / $20.63\%H$
Failure-inclusive point-level P90	$61.42\%H$
$\mathrm{LR}_{0.3}$	$143/203 = 70.4\%$ $[60.9\%, 79.8\%]$
$\mathrm{LR}_{0.4}$	$120/203 = 59.1\%$ $[48.5\%, 69.6\%]$
$\mathrm{LR}_{0.5}$	$93/203 = 45.8\%$ $[35.2\%, 56.3\%]$

O Fixed-Denominator Error Propagation

For 61 questions from 16 clean panels, calibration, solver, scorer, and denominator are fixed while the curve representation changes. Raw L1 coordinates provide the same-source solver ceiling.

Table O.1: Error propagation with a fixed denominator. Medians use available answers; missing, non-finite, or wrong-cardinality outputs fail Pass@5% and receive 100% capped error.

Curve representation	Answered	Median error	Pass@5%	Capped mean
Raw manual L1 coordinates	61/61	0.00%	61/61	0.00%
GT mask $\rightarrow$ reconstructed centerline	61/61	0.79%	47/61	7.65%
Complete automatic system	60/61	3.34%	35/61	16.11%

Stage-wise Pass@5% counts are $61 \rightarrow 47 \rightarrow 47 \rightarrow 45 \rightarrow 35 \rightarrow 35$ for manual coordinates, GT-mask reconstruction, axis/legend handling, foreground extraction, curve separation, and predicted K, respectively. One-at-a-time substitutions for curve count, foreground, and separation increase capped error by 12.68 [4.82, 21.72], 12.55 [1.36, 25.31], and 19.79 [5.56, 34.72] percentage points, respectively (paired panel-bootstrap 95% intervals). These marginal effects are not additive.

P Additional Evaluation and Operating Boundaries

Verifier replay. We replay fixed outputs, prompts, and images on 60 blinded-screened positive panels and 40 original L2 curation rejects. Acceptance requires curve-count, shape, and axis-range agreement; noise/artifact flags are advisory.

Table P.1: Verifier replay under the common three-criterion rule; reference labels come from screening.

Verifier	Positive panels retained	Curation rejects rejected
Qwen2.5-VL-7B-Instruct (local)	55/60	17/40
Qwen-VL-Max	46/60	39/40
Gemini-3.1-Pro-preview	28/60	40/40
GPT-5.4	23/60	39/40

Table P.2: Curve-count source sensitivity on clean-90. Foreground masks, separation, matching, and scoring remain fixed.

Curve-count source	Exact K	Complete-pipeline $\mathrm{LR}_{0.5}$
Gemini-3.1-Pro-preview	$75/90$	$93/203 = 45.81\%$
Qwen2.5-VL-7B-Instruct (local)	$72/90$	$93/203 = 45.81\%$
Qwen-VL-Max	$71/90$	$96/203 = 47.29\%$
Qwen3-VL-Plus	$70/90$	$94/203 = 46.31\%$
GPT-5.4	$66/90$	$91/203 = 44.83\%$
Deterministic silhouette-K	$22/90$	$69/203 = 33.99\%$

Workflow screening yield. The 1,000-image audit records runtime availability and quality screening separately. Quality screening assigns each rejection to its first failed criterion; final Qwen-VL-Max review retains $707/1{,}000$ images (70.7%).

Table P.3: Cumulative runtime availability and quality-screening retention in the same 1,000-image audit, using separate stage criteria.

Stage	Operational chain	Quality screening
Audit entry	1,000	1,000
Axis geometry	999	920
Axis/tick range	999	841
Legend association	999	783
Line reconstruction	998	707
Final Qwen-VL-Max screening	707	—

Fixed-model representation control. Across 87 questions from 26 single-curve panels, only image and trajectory inputs vary; GPT-5.4, questions, prompt wrapper, and scorer are fixed. Pass@5% uses all questions; medians use parseable answers.

Table P.4: Five-input representation control with one fixed GPT-5.4 model on 87 questions.

Input	Numeric answers	Median relative error	Pass@5%
Image only	$87/87$	19.4%	32.2%
Image + manual L1	$87/87$	1.1%	62.1%
Manual L1 only	$87/87$	0.8%	66.7%
Image + automatic L2	$87/87$	4.3%	54.0%
Automatic L2 only	$86/87$	5.4%	48.3%

Versus image only, median-error differences are -18.3 pp $[-26.5, -12.0]$ with manual L1 and -15.1 pp $[-22.2, -9.2]$ with automatic L2 (paired panel-bootstrap 95% intervals). Data-only versus image-plus-data intervals include zero for both sources.

Strict calibration policy. Synthetic references provide relative coordinates; strict numeric calibration abstains from absolute physical-unit output without numeric ticks.

Dense specialized-model control. Public zero-shot checkpoints are compared on 16 constrained-argmax, 16 FWHM, and two Raman peak-list questions, with visible positional targets. From 54 archived questions, we exclude 16 without numeric y ticks, three offset-sensitive ratios, and one incomplete peak list. Missing/nonnumeric outputs fail Pass@5%; medians use parseable answers.

Table P.5: Zero-shot comparison on 16 dense multi-curve panels and 34 questions.

System	No-answer	Median relative error	Pass@5%
MatCha [38] (image)	$0/34$	93.7%	$3/34$
UniChart-ChartQA [39] (image)	$18/34$	89.0%	$2/34$
MatDeplot trace + deterministic calculation	$1/34$	4.6%	$17/34$

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract enumerates three contributions: **MatCurvs** (a three-tier benchmark with L0 routing, L1 manual pixel polylines, and L2 screened real-coordinate curves; §4), **MatDeplot** (a local extraction pipeline; §5), and **MatCurvs-Reasoning** (a 14,740-question deterministic numerical-reasoning evaluation; §7). Each is empirically substantiated in the corresponding section, with the pixel-anchoring headline in §6 ($\mathrm{LR}_{0.5}$=45.8% vs. $\leq$1.2% for hosted VLMs) and the end-to-end agent pilot in §7 (19 derived-parameter tasks: 19/19 vs. image-only VLM 2/19).

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: §8 discusses the scope of the tested materials-chart tasks, remaining pixel-reconstruction errors, and model choices and calibration requirements. Local curve-count alternatives and verifier comparisons are reported in Appendix P.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper presents an empirical pipeline, an annotation protocol, and a four-method causal evaluation; it contains no theorems, propositions, or proofs.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.

- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the extraction implementation, training recipes, and benchmark data. Sections 3 to 5 specify the evaluation protocol, data construction, and extraction procedure, with pseudocode in Algorithm 1. Training settings, the leakage audit, and additional controls are detailed in Appendices D, L and P.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The dataset and its documentation are publicly available on Hugging Face. The extraction code, configuration files, training recipes, installation instructions, and usage examples are available on GitHub.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: §5 and Algorithm 1 specify the architecture, inference thresholds, augmentation, curve separation, and fallback settings. §4 describes the benchmark and curation protocol; Appendix L documents the n=90 leakage-clean hold. Appendix D gives the optimiser, training schedule, and architecture/data/loss sweep across 24 configurations and 25 checkpoint rows.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Appendix H reports paired image-cluster bootstrap CIs (10,000 resamples) for IoU and coverage metrics. Appendix N reports panel-bootstrap intervals for point support and line recall, and Table P.4 and its accompanying analysis report the fixed-model representation comparison and paired panel-bootstrap intervals.

Guidelines:

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests.
- The method for calculating the error bars should be explained.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The foreground model was trained on one NVIDIA H100 (80 GB), with training settings in Appendix D. Table I.1 reports median inference latency of 1.5 s/image on this hardware. Per-image hosted-API costs are reported separately in Table 2.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics?

   Answer: [Yes]

   Justification: We have reviewed the Code of Ethics and confirm full conformance. The MatCurvs benchmark is built from open-access materials-science published-figure crops; no human subjects, no PII, no dual-use content. The MatCurvs-Reasoning question/answer pairs are deterministic transformations of L2 (no human-rated labels).

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The conclusion discusses easier reuse of published measurements in materials research. §8 and Appendix O discuss extraction and calibration errors that can affect downstream calculations. Confidence scores and abstention mechanisms for flagging uncertain outputs are described in §5.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse?

    Answer: [N/A]

    Justification: The released checkpoint is a chart-foreground segmenter that emits pixel masks of plotted curves; it has no generative output and no dual-use risk.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Third-party software and models are credited in Sections 2 and 5 and retain their original licenses and service terms. YOLO11 uses AGPL-3.0. Source chart panels retain the licenses of their original publications, as stated in the dataset documentation.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: The Hugging Face dataset cards describe the MatCurvs tiers, annotation formats, construction, and reasoning question sets. The GitHub README and training guide document installation, inputs and outputs, model requirements, and training configurations. Code is released under MIT and the new annotations under CC-BY-4.0.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The L1 manual pixel-polyline annotations on 204 panels were produced by a single trained annotator using a WebPlotDigitizer-derived tool, with a second annotator independently re-annotating 20 panels for inter-annotator consistency and a domain expert reviewing every panel (§4.2). All annotators are co-authors; no external crowdsourcing was used and no compensation was paid.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The study involves no human subjects; no IRB approval is required.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research?

Answer: [Yes]

Justification: LLM tools were used to assist in drafting the initial manuscript.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.