Project Report

“Natural Language Processing (W2526) ”

Submitted by

Umang Dholakiya

Supervised by

Prof. Dr. Patrick Levi

Ostbayerische Technische Hochschule Amberg-Weiden
Department of Electrical Engineering, Media and Computer Science

December 22, 2025

Contents

Abstract

Image-to-text models are commonly trained on photographic images paired with short captions and therefore perform poorly on schematic representations, which are largely absent from standard training datasets. Quantum circuit diagrams are a representative example of this limitation, as they encode structured, symbolic information using domain-specific visual conventions rather than natural imagery. To enable fine-tuning of image-to-text models for this domain, a dedicated dataset of quantum circuit images paired with structured semantic metadata is required. This project investigates whether compiling such a dataset is feasible with reasonable effort by automatically collecting figures from scientific publications on the arXiv platform, focusing on recent papers in the *quant-ph* category, while accounting for the fact that not all publications in this category are directly related to quantum computing. The task is complicated by unstructured PDF layouts, the coexistence of circuit and non-circuit figures, and frequent extraction errors produced by naive figure-cropping approaches. To address these challenges, a reproducible end-to-end pipeline was developed that integrates visual layout analysis to extract complete figure regions without fragmentation or text leakage, followed by a hybrid filtering strategy. This strategy combines a convolutional neural network–based visual classifier with natural language processing techniques applied to captions and surrounding text, as well as information extraction methods to identify domain-specific entities such as quantum gates and algorithm references. The pipeline outputs validated quantum circuit images together with rich JSON metadata, provenance information, and detailed logs, resulting in a prototypical dataset of over 250 circuit diagrams and demonstrating that a specialized, automated approach can bridge the data gap for schematic image-to-text tasks in the quantum computing domain.

1 Introduction

Recent image-to-text and vision–language models perform well on natural images, but they generalize poorly to schematic and symbolic graphics. Quantum circuit diagrams are a representative example: their meaning is encoded by structured symbols (wires, gates, controls, measurements) and spatial/temporal layout rather than texture or color. Because such diagrams are largely absent from common vision–language training datasets, off-the-shelf models typically fail to generate accurate or useful descriptions for quantum circuits.

A practical way to address this gap is to fine-tune image-to-text models on a domain-specific dataset of quantum circuit images paired with reliable textual descriptions. However, constructing such a dataset is non-trivial. Quantum circuits appear inside scientific PDFs alongside many non-circuit figures (plots, tables, block diagrams), and PDF layouts vary substantially across papers. Naïve extraction methods such as page rasterization or simple cropping often produce incomplete figures, include surrounding text, or extract irrelevant graphics.

This project focuses on recent papers from arXiv, primarily in the *quant-ph* category, and proposes a reproducible end-to-end pipeline to automatically construct a quantum circuit image-to-text dataset. The pipeline aims to (i) extract complete circuit figures, (ii) reject visually similar non-circuit figures, and (iii) attach structured metadata (e.g., source paper identifiers, figure indices, extracted gate tokens, and caption-aligned descriptions) suitable for downstream training and analysis. The emphasis is on precision, traceability, and reproducibility rather than maximizing extraction volume, enabling systematic evaluation and error analysis under realistic conditions.

1.1 Problem Statement and Objectives

The objective of this project is to assess whether a domain-specific image-to-text dataset for quantum circuit diagrams can be compiled from scientific literature with reasonable effort, and to deliver a small prototypical dataset suitable for downstream model fine-tuning. Concretely, the dataset must contain (i) quantum circuit images extracted from scientific PDF papers and (ii) structured metadata that supports semantic interpretation and traceability, including provenance (paper identifier, page number, and figure region) and text alignment (caption and relevant surrounding context).

Formally, the input to the system is a predefined list of arXiv identifiers drawn from recent years of the *quant-ph* category, together with configuration parameters controlling acquisition and processing. The required output is a reproducible dataset consisting of validated circuit images and accompanying JSON records, as well as logs and intermediate artifacts that allow an examiner to reproduce the results and diagnose failure modes. The central technical problem is that scientific PDFs are not standardized: figure regions may be composed of vector graphics, raster images, or mixtures; multi-panel figures are common; and the same paper typically contains a various set of non-circuit figures (plots, tables, architectural diagrams, photographs) that must be rejected. A second difficulty is that the *quant-ph* category is broader than quantum computing, so a large fraction of papers contain no quantum circuits at all, which makes selective extraction essential.

To address this problem, the project pursues the following objectives:

1. **Reproducible paper acquisition.** Acquire PDFs from arXiv in a controlled and repeatable order, apply rate limiting, and record processing status to ensure deterministic reruns.

2. **Robust figure extraction.** Detect and extract complete figure regions without page screenshots, partial crops, or inclusion of surrounding body text; handle multi-panel figures by extracting the full composite figure rather than isolated sub-panels.

3. **High-precision circuit filtering.** Identify quantum circuit diagrams with a strict acceptance criterion to minimize dataset pollution (false positives), employing a hybrid filtering strategy that is robust to visual variation across publications.

4. **Text alignment and semantic metadata.** Associate each accepted circuit image with caption and context text and extract domain-specific entities (e.g., quantum gates and algorithm mentions) into structured metadata fields.

5. **Traceability and debugging support.** Store intermediate outputs (e.g., rejected candidates with reason codes) and detailed logs so that extraction errors and classification failures can be analyzed systematically.

Success is defined in terms of dataset usability and reproducibility rather than raw extraction volume. A successful pipeline produces circuit images that are complete (not fragmented or cut), minimally contaminated by surrounding text, and accompanied by metadata sufficient to trace each sample back to its source document and justify why it was accepted. Where full automation fails, the system must expose the failure mode explicitly through logs and debug artifacts, enabling targeted improvements rather than silent degradation.

2 Data and Materials

This project constructs a quantum-circuit dataset directly from raw arXiv PDFs (no pre-existing circuit corpus) [1]. The key materials are the fixed paper list defining the search space and the exported dataset artifacts (PNG images, JSON metadata, and a per-paper CSV summary).

2.1 Source Data and Acquisition

For Exam ID 37, the pipeline processes a fixed list of 26,908 arXiv identifiers from the quant-ph category (paper_list_37.txt) in a deterministic order. PDFs are downloaded automatically from arXiv under a 3-second rate limit; no manual selection or hand-filtering is performed. The corpus functions as a candidate pool (many papers contain no circuits), and zero-yield papers are retained for complete accounting.

2.2 Extraction and Filtering Constraints

Figure candidates are extracted with layout-aware methods that target complete figure regions: partial crops and text-contaminated regions are rejected, and multi-panel figures are extracted as a full composite region. Since most figures are non-circuits (plots, tables, general diagrams), filtering is conservative and combines visual structure cues with caption/context text signals, prioritizing precision to minimize dataset pollution.

2.3 Dataset Outputs and Intermediate Artifacts

The exported dataset consists of three coupled deliverables: (i) lossless circuit crops stored in images_37/, (ii) one JSON record per image stored in output/ (provenance, aligned caption/context, extracted entities such as gate mentions and algorithm/problem cues), and (iii) paper_list_counts_37.csv reporting extracted circuit counts per paper, including zeros. For auditability, intermediate outputs are preserved, including rejected candidates with provenance fields and rejection reason codes, enabling inspection of failure modes without reprocessing PDFs.

3 Methodology: End-to-End Dataset Construction Pipeline

The proposed system is implemented as a modular, multi-stage pipeline (Figure 1). Each stage performs a specific transformation step, progressively refining raw scientific PDFs into validated quantum circuit images paired with structured metadata. The staged design enables precise error isolation, supports reproducibility, and allows conservative filtering decisions to be justified and audited through intermediate artifacts.

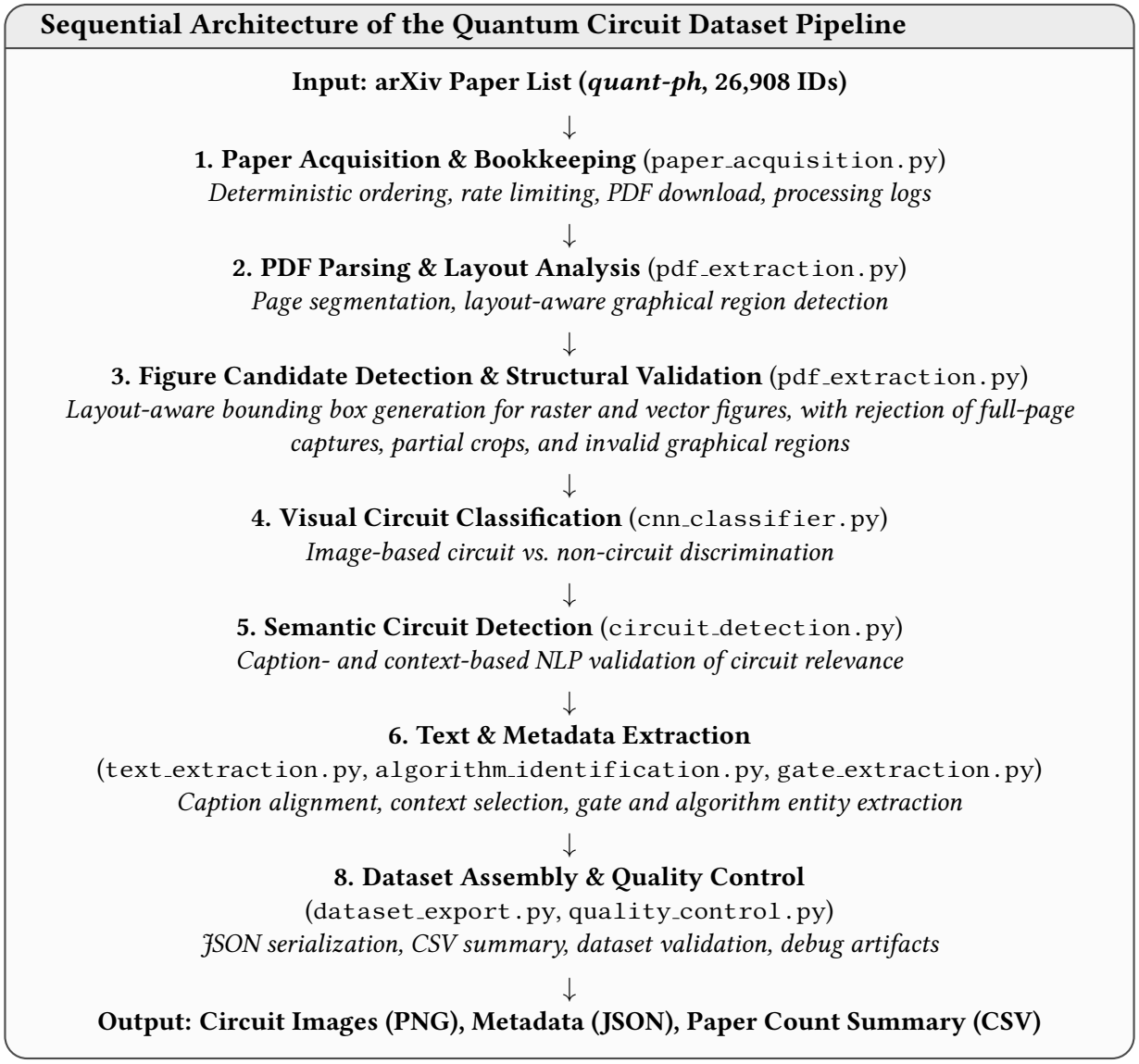


Figure 1: *End-to-end pipeline for automated extraction and curation of quantum circuit datasets from scientific PDFs.*

3.1 Paper Acquisition and Bookkeeping

This stage provides a controlled and reproducible input stream of scientific papers. It ensures deterministic traversal of a predefined arXiv paper list, reliable PDF acquisition under explicit rate limits , and persistent bookkeeping of paper-level outcomes. By recording both successful and failed acquisitions, the pipeline enables systematic analysis of dataset coverage and sparsity.

3.1.1 Deterministic Ordering and Reproducibility

Paper acquisition follows a fixed list of arXiv identifiers processed sequentially by src/paper_acquisition.py. This guarantees that each pipeline execution processes the same papers in the same order. For every paper, processing status and extracted circuit counts are stored in a persistent CSV checkpoint file (paper_list_counts_37.csv), ensuring repeatable paper-level results across runs.

Dynamic discovery methods such as keyword-based arXiv queries were intentionally avoided, as they introduce temporal variability and topical bias. Using a static paper list provides a stable, transparent input stream and allows unambiguous traceability from each dataset entry back to its source document.

3.1.2 Rate Limiting and Failure Handling

PDFs are downloaded using the official arXiv API with an explicit, configurable delay between requests (see `PDFDownloader._wait_for_rate_limit()` in `paper_acquisition.py`)x(default: 3 seconds) to prevent throttling and request blocking. Download failures are handled explicitly: transient errors trigger bounded retries, while persistent failures are logged and retained in the checkpoint file with zero extracted circuits.

More aggressive strategies such as parallel downloads were deliberately avoided. Although faster, they increase the risk of partial failures and inconsistent results. The selected sequential, rate-limited approach prioritizes robustness and reproducibility over throughput, aligning with the project's focus on dataset quality and methodological reliability.

3.2 PDF Parsing and Layout Analysis

This stage parses each arXiv PDF into a page-wise representation and detects candidate figure regions using layout-aware cues. The goal is to extract visually complete figure candidates with minimal contamination from surrounding text, despite various PDF encodings and figure representations [6].

3.2.1 PDF Representation and Page Processing

Each document is processed page-by-page using a PDF-native parser that preserves page geometry and access to both text and graphical content. For every page, the pipeline records the page index, geometric extent, and available content signals, enabling deterministic iteration and precise provenance tracking at the level of paper ID, page number, and bounding box.

The parser produces two outputs used downstream: (i) bounded figure candidates that can be rendered to PNG at a controlled resolution, and (ii) lightweight textual structure from nearby text blocks for attaching caption candidates. The output of this stage is therefore a list of figure candidates per paper, each with image data, page number, bounding box coordinates, and optional caption text.

3.2.2 Layout-Aware Graphical Region Detection

Figure regions are detected using a hybrid approach that combines document layout analysis with PDF-native graphics extraction. Visual layout analysis is applied to rendered pages to detect complete figure regions, mitigating common failure modes of naïve extraction such as fragmented vector diagrams and text-contaminated crops.

To account for diverse PDF encodings, the pipeline augments layout detection with embedded image extraction and vector-region rendering. Vector drawing primitives are clustered into composite regions, which is essential for quantum circuits typically encoded as collections of small vector

paths. Candidate regions are rendered at a fixed resolution and filtered using explicit size and aspect-ratio constraints to suppress full-page captures and small artifacts.

Stability and scalability are ensured by skipping pages with excessive vector complexity and deduplicating overlapping candidates using spatial overlap and image-content hashing, retaining the most complete region. Alternative approaches based solely on embedded images or page-level rasterization were rejected due to missed vector figures and frequent partial or text-contaminated extractions. The adopted hybrid strategy provides controlled bounding boxes, preserves vector content, and supports reproducible dataset construction.

3.3 Figure Candidate Detection and Structural Validation

This stage extracts candidate figure regions from each PDF page and applies lightweight structural validation to remove obvious extraction artifacts. Candidate generation and validation are implemented in `src/pdf_extraction.py`. Key entry points are:

- `LayoutBasedFigureExtractor.extract_all_figures()`
- `FigureExtractor.extract_all_figures()`

3.3.1 Candidate Generation and Structural Filtering

Input: a downloaded PDF file.

Processing: pages are scanned for (i) embedded raster images and (ii) vector-based graphical regions that must be rendered to pixels. Raster figures are extracted directly via `LayoutBasedFigureExtractor._extract_embedded_images()`, while vector figures are identified from page drawing primitives and rendered at a fixed DPI via `LayoutBasedFigureExtractor._extract_vector_figures()`. DPI and geometric thresholds (min/max width/height, aspect ratio bounds) are defined in `src/config.py` under `ExtractionConfig`. During extraction, simple geometric constraints are applied to discard degenerate regions such as extremely small fragments or full-page sized captures (see `LayoutBasedFigureExtractor._check_size_constraints()`).

Multi-panel figures are handled by grouping spatially adjacent graphical components into a single composite region prior to rendering:

- `LayoutBasedFigureExtractor._cluster_drawings_to_figures()`
- `LayoutBasedFigureExtractor._cluster_bboxes()`
- `LayoutBasedFigureExtractor._merge_nearby_clusters()`

After candidate generation, overlapping regions on the same page are deduplicated using spatial overlap heuristics, retaining the most complete (largest-area) crop:

- `LayoutBasedFigureExtractor._deduplicate_figures()`
- `LayoutBasedFigureExtractor._figures_overlap()`

Output: a list of structurally valid figure candidates, each associated with a page number and bounding box coordinates. All extracted candidates are saved to `phase1_raw_figures/` (see `save_figure_to_png()` in `src/pdf_extraction.py` and its usage in `QuantumCircuitDatasetPipeline._process_paper()` in `src/main.py`).

3.3.2 Debug Artifacts and Failure Analysis

To support systematic error analysis, all extracted candidates are retained prior to semantic filtering (see `QuantumCircuitDatasetPipeline._process_paper()` in `src/main.py`). During later stages, candidates rejected by the visual classifier are copied into `rejected_figures/` using the same filename convention (see `QuantumCircuitDatasetPipeline._create_circuit_entry()` in `src/main.py`, which applies `CNNCircuitClassifier.analyze()` from `src/cnn_classifier.py` and routes outputs via `CONFIG.paths.rejected_images_dir`).

This produces an auditable trail of extraction outcomes, allowing inspection of common failure modes such as cropping errors, fragmentation, and duplicate detections without re-downloading PDFs, and enabling threshold tuning based on concrete rejected examples.

3.4 Hybrid Vision–Language Filtering

As established in the Problem Statement, the *quant-ph* category is dominated by non-circuit figures, resulting in extreme sparsity of relevant quantum circuit diagrams. To prevent dataset pollution while maintaining robustness across various PDF layouts and figure styles, the pipeline adopts a conservative *hybrid vision–language gatekeeper* strategy. In this design, a figure is included in the dataset only if it satisfies both visual plausibility and semantic relevance criteria. This strict conjunction ensures high precision, which is critical given the infeasibility of large-scale manual validation.

3.4.1 Visual Circuit Classification

The first filtering component performs image-based discrimination using a convolutional neural network implemented in src/cnn_classifier.py. Its responsibility is to reject visually implausible figure candidates—such as plots, experimental diagrams, or hardware images—before semantic text analysis is applied.

Input: rasterized figure images produced by the PDF extraction and structural validation stage. These images correspond to complete figure regions and are free of obvious extraction artifacts.

Processing: each candidate image is resized to a fixed resolution of 224×224 pixels and passed through a fine-tuned **ConvNeXt-Tiny** model [5]. The architecture employs large 7×7 convolutional kernels in its stem, enabling the network to capture extended spatial context that is characteristic of quantum circuit layouts, such as long horizontal wires, gate alignments, and control–target structures. The classifier head maps a 768-dimensional feature representation to two output classes (*circuit* and *non-circuit*), producing a probabilistic confidence score based purely on pixel-level information.

The model was fine-tuned using two complementary datasets: (i) a curated subset of 2,136 unique quantum circuit images sampled from a publicly available quantum circuit image dataset [3], and (ii) 2,245 randomly selected non-circuit figures drawn from the DocFigure dataset [4], which contains diverse scientific diagrams, plots, and schematics. This combination exposes the classifier to both circuit-specific visual patterns and realistic negative examples encountered in scientific PDFs, improving generalization under real-world document variability.

Design Rationale: rule-based visual filtering approaches (e.g., edge density thresholds, line detection, OCR-based text suppression) were considered but rejected due to brittleness across different PDF renderings and diverse circuit drawing conventions [2]. Such heuristics frequently fail on vector-rendered figures, stylized gate symbols, or dense multi-panel layouts. End-to-end multimodal models combining image and text inputs were also deemed unsuitable, as they increase system complexity, reduce interpretability, and tightly couple visual and semantic reasoning, complicating debugging and controlled error analysis. In contrast, a CNN-based visual classifier provides a robust,

learned representation that is modular, scalable, and reproducible, while remaining computationally efficient for large-scale processing.

Output: the classifier produces a visual confidence score $P_{\text{vis}} \in [0, 1]$. A threshold of 0.5 is applied to filter out visually obvious non-circuits prior to semantic verification. Crucially, this stage does *not* make a final acceptance decision; it produces a visual signal that is passed downstream and combined with semantic evidence.

3.4.2 Semantic Text Verification

Visual appearance alone is insufficient to reliably distinguish quantum circuits from visually similar block diagrams, such as neural network architectures or control flow schematics. Therefore, a secondary semantic filter is applied using the `QuantumCircuitDetector()` implemented in src/-circuit_detection.py.

This module analyzes figure captions and surrounding textual context using a deterministic weighted keyword and pattern matching strategy. High-confidence circuit indicators (e.g., "quantum circuit", "gate sequence", explicit gate names) contribute strong positive evidence, while terms characteristic of result plots or experimental figures (e.g., "error rate", "fidelity vs.", "histogram") impose explicit penalties. This negative-context suppression is essential for rejecting figures that pass the visual filter but are semantically unrelated to quantum circuit diagrams.

The output of this stage is a semantic confidence score $C_{\text{text}} \in [0, 1]$ reflecting the textual support for the circuit hypothesis.

3.4.3 Consensus Logic

A figure is included in the final dataset if and only if it satisfies the conjunction of both visual and semantic criteria[7]:

$$\text{Accept}(x) \iff \big(P_{\text{vis}}(x) > 0.5\big) \wedge \big(C_{\text{text}}(x) \geq 0.35\big) \tag{1}$$

This strict "AND" logic enforces high precision by requiring agreement between independent visual and textual signals. Figures that satisfy only one modality are rejected, ensuring that visually plausible but semantically irrelevant figures—and vice versa—do not contaminate the dataset. This hybrid filtering strategy directly addresses the observed sparsity and heterogeneity of scientific PDFs while preserving reproducibility, modularity, and controlled error analysis.

3.5 Semantic Circuit Detection

This stage performs caption- and context-based semantic verification to reduce false positives that remain after visual filtering. Because the CNN cannot reliably separate true circuits from visually similar block diagrams or experimental schematics, a deterministic text-based detector (implemented in src/circuit_detection.py) checks whether the surrounding text provides explicit circuit evidence and suppresses candidates whose language indicates non-circuit content.

Input: (i) the figure caption and (ii) a small set of surrounding context snippets aligned to the figure.

Output: a structured result containing a circuit/non-circuit decision, a semantic confidence score $C_{\text{text}} \in [0, 1]$, and an evidence list of matched terms/patterns to support auditability.

3.5.1 Deterministic Evidence Scoring and Decision Rule

Semantic detection uses weighted keyword and pattern matching over both caption and context. Circuit-indicative phrases (e.g., "quantum circuit", "gate sequence", explicit gate names) contribute

positive evidence, while plot- and evaluation-oriented language (e.g., “histogram”, “error rate”, “fidelity vs.”) contributes explicit negative penalties. Caption and context are scored separately and combined via weighted fusion (0.6 caption, 0.4 context). A figure is accepted if the fused score exceeds a configurable threshold (default 0.35).

3.5.2 Rationale and Observed Fix

A deterministic strategy was chosen over learned semantic classifiers to preserve reproducibility, computational efficiency, and transparent debugging via explicit evidence lists. The main observed semantic failure mode was visually circuit-like figures whose text described plots or architectural schematics; this was mitigated by introducing strong negative-context penalties and non-circuit caption patterns, which reduced false positives while keeping the decision logic auditable.

3.6 Metadata Enrichment, Dataset Export, and Quality Control

After figure validation, the pipeline (i) enriches each accepted circuit image with aligned text and semantic entities, (ii) serializes records into submission-ready dataset artifacts, and (iii) applies post-assembly quality control. These steps are descriptive and reproducible: they do not change figure acceptance, but they standardize metadata, preserve provenance to the source PDF, and ensure the exported dataset is structurally consistent.

3.6.1 Text Alignment and Entity Extraction

Input: validated circuit images and PDF text signals (figure captions and nearby text blocks).

Processing: captions are aligned to figures using spatial and reference-based cues (caption-first strategy; src/text_extraction.py). When captions are insufficient, a conservative subset of surrounding context is checked: only blocks that reference the figure or fall within a bounded spatial/text neighborhood are included, avoiding the failure mode where large text windows introduce unrelated narrative content. Captions and context are stored as separate segments to preserve granularity for debugging and supervision.

From the aligned text, the pipeline extracts (i) optional algorithm/protocol labels using conservative pattern-based recognition (src/algorithm_identification.py), leaving the field empty when evidence is weak, and (ii) gate mentions (e.g., H, X, CNOT, RX) normalized to a canonical vocabulary to resolve aliases (src/gate_extraction.py). This provides semantic cues without attempting full executable circuit reconstruction.

3.6.2 Dataset Export, Bookkeeping, and Quality Control

Output artifacts: (i) one JSON record per accepted circuit image written to output/ (dataset export in src/dataset_export.py), (ii) extracted images stored in images_37/, and (iii) paper_list_counts_37.csv reporting accepted circuit counts per paper, including zero-yield papers for complete accounting and coverage analysis.

Quality control: after assembly, a lightweight validation layer (src/quality_control.py) checks required metadata fields, detects duplicate identifiers and degenerate entries, and performs basic sanity checks on extracted entities. Failed entries are not silently discarded; they are logged and preserved as debug artifacts. Rejected or flagged figures are stored separately using the same naming convention as accepted images, enabling targeted inspection and threshold tuning. Two recurring issues handled by this stage are incomplete metadata (missing captions/sparse context) and duplicate/near-duplicate figures caused by overlapping extraction regions; QC preserves an auditable trail from raw PDFs to final dataset artifacts.

3.7 End-to-End Execution

The pipeline runs sequentially with explicit intermediate artifacts for auditability. A typical run is invoked as:

command: python run_pipeline.py –target 250

run_pipeline.py → main.py → paper_acquisition.py → pdf_extraction.py → cnn_classifier.py → circuit_detection.py → text_extraction.py → dataset_export.py

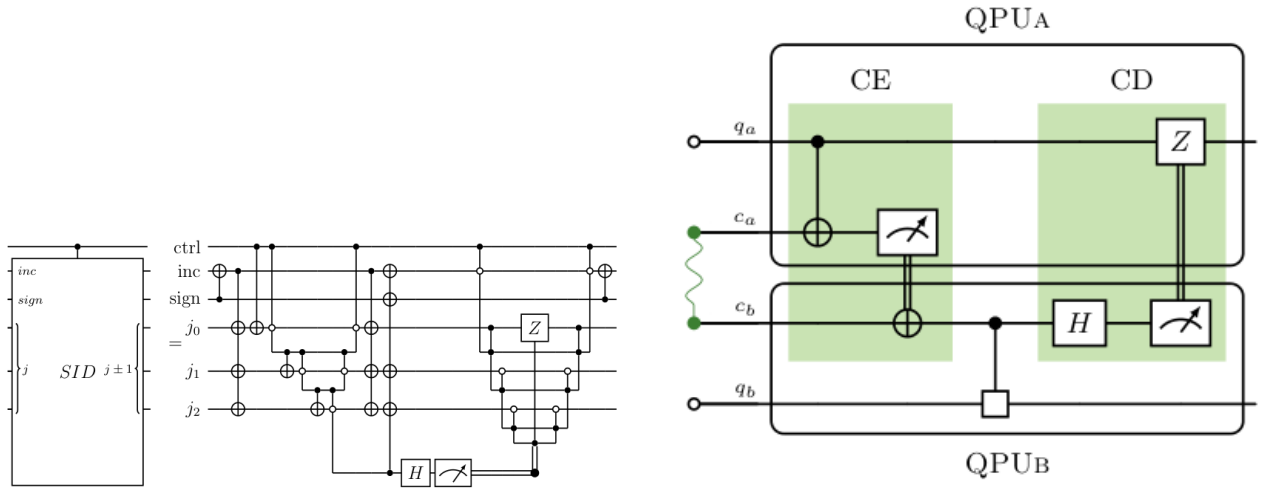
Intermediate outputs, including raw figure candidates and rejected samples, are preserved in dedicated directories (e.g., phase1_raw_figures/, rejected_figures/) to support systematic error analysis without re-downloading PDFs.

4 Experiment and Results

This section evaluates the pipeline output using representative qualitative examples (one success case and one failure case) and a manual quantitative check of the extracted dataset. The goal is to assess (i) whether the pipeline correctly detects quantum circuit figures, (ii) whether the extracted gate tokens and metadata are reliable, and (iii) where the dominant failure modes occur in practice.

4.1 Success Case: Correct Circuit Detection and Metadata Extraction

Figure 2a (arXiv:2405.10416, page 18, Fig. 6) shows a successful end-to-end extraction. The pipeline correctly links the image to its source (`arxiv_id`, page, and figure index), extracts a valid set of gate primitives {CNOT, H, Z, MEASURE}, and aligns the circuit with nearby caption/context text. Based on this text, the entry is assigned `quantum_problem = "Oracle Implementation"`, consistent with the caption describing an optimized oracle O_H. Provenance is preserved through stored `text_positions`, enabling verification of the supporting sentences.



(a) *Success example (`2405.10416_18_6.png`): correct circuit detection, gate extraction, and caption-aligned task labeling (oracle implementation).*

(b) *Failure example (`2507.01090_9_8.png`): circuit detected, but gate extraction over-includes generic labels and the caption lacks explicit algorithm cues, leading to an unspecified task label.*

Figure 2: *Qualitative examples of pipeline output (left: success case, right: failure case).*

4.2 Failure Case: Gate Over-Inclusion and Missing Task Identification

Figure 2b (arXiv:2507.01090, page 9, Fig. 8) illustrates a typical partial failure and represents a broader pattern observed across many papers. The circuit is correctly detected and several standard gates are recovered, but the extracted gate list includes over-general labels (e.g., `U` for unknown boxed operations and occasional spurious `I`), reducing fidelity for reconstruction. More importantly, the semantic module cannot reliably infer an algorithm or problem type from the surrounding text. In this example (and many similar cases), the caption references an operator definition (Eq. B1) or protocol-specific terminology rather than an explicit named algorithm. As a result, the pipeline defaults to `quantum_problem = "Unspecified quantum circuit"`. When algorithm detection fails, the system falls back to storing the raw caption text in the `descriptions` field so that each circuit still retains a human-authored natural-language description, even if the task label remains coarse.

4.3 Manual Evaluation Summary

To quantify overall quality, a manual inspection was performed on a sample of 250 extracted images by comparing the stored metadata and the extracted figure content against the corresponding source PDFs. Out of these 250 candidates, 170 were confirmed as correct quantum circuit extractions, while the remaining 80 were false positives.

A major source of false positives is the vector-graphics extraction stage: some non-circuit figures (or fragments of complex vector diagrams) are reconstructed into circuit-like line structures that trigger the detector. This inflates recall but reduces precision, especially for papers where figures contain dense vector primitives (e.g., mixed diagrams, protocol schematics, block diagrams, or annotated layouts). In addition, even when the extracted image is a true quantum circuit, metadata quality is sometimes only partially correct. Typical issues include over-general gate tokens (e.g., labeling unknown operations as `U`), missed multi-controlled structure, and missing or overly coarse task labels when captions do not contain explicit algorithm keywords. These observations motivate the error analysis in the next section and highlight that the pipeline currently produces a mixture of fully correct entries, partially correct metadata, and non-circuit false positives.

5 Dataset Quality Assessment and Error Analysis

This section critically assesses the quality of the constructed dataset by analyzing observed error modes, limitations, and design trade-offs encountered during development. The goal is not to claim perfect extraction, but to demonstrate that dataset construction decisions were made deliberately, based on empirical observation and task constraints, and that the final pipeline represents the most defensible solution among the evaluated alternatives.

5.1 Initial Approaches and Observed Failure Modes

Several straightforward approaches were evaluated during early development and were ultimately rejected due to systematic failure modes.

A first attempt relied on direct PDF image extraction (embedded raster images only). While computationally simple, this approach failed to recover the majority of quantum circuit diagrams, which are frequently encoded as vector graphics composed of many small drawing primitives. As a result, recall was unacceptably low and biased toward a narrow subset of papers.

A second approach used full-page rasterization followed by heuristic cropping. This strategy increased recall but introduced severe quality issues: partial crops, inclusion of surrounding body

text, loss of resolution, and fragmentation of multi-panel figures. Empirical inspection showed that many extracted images were unusable for downstream learning despite appearing visually plausible at first glance.

Finally, purely text-driven filtering (accepting figures whose captions contained keywords such as "quantum circuit") was evaluated. This approach exhibited extremely high false-positive rates, admitting plots and experimental figures whose captions mentioned "circuits" only tangentially (e.g., performance plots of circuit depth or error rates). These failures demonstrated that neither vision-only nor text-only strategies were sufficient in isolation.

5.2 Error Modes in the Final Pipeline

Even in the final pipeline, several residual error modes remain. These are documented explicitly rather than hidden.

False negatives. Some valid quantum circuits are rejected because they lack explicit textual indicators in captions or surrounding context, or because they are visually stylized in ways not well represented in the training data. This is an acknowledged limitation and a direct consequence of prioritizing precision over recall. Given the task goal of dataset quality rather than completeness, this trade-off was considered acceptable.

Borderline semantic ambiguity. Certain figures occupy a gray area between circuit diagrams and abstract block schematics (e.g., high-level compilation pipelines or architectural diagrams). In these cases, the pipeline tends to reject ambiguous figures unless both visual and textual evidence strongly support the circuit hypothesis. This conservative behavior reduces dataset pollution but may exclude some conceptually relevant diagrams.

Residual structural artifacts. Despite layout-aware extraction, a small number of figures exhibit minor cropping imperfections or excess whitespace. These cases are rare and are preserved as debug artifacts for transparency rather than silently removed.

5.3 Design Decisions and Justifications

Each major design decision in the final pipeline directly addresses failures observed in earlier attempts.

Hybrid vision–language filtering. The decision to require agreement between a visual classifier and semantic text verification was motivated by empirical evidence that single-modality filters consistently fail in this domain. Visual classifiers alone misclassify plots and block diagrams; text-based methods alone are vulnerable to misleading captions. The conjunction rule explicitly targets these complementary weaknesses and demonstrably reduces false positives.

Deterministic semantic scoring over learned NLP models. Transformer-based text classifiers were considered but rejected. In practice, they introduced opaque decision boundaries, unstable behavior across papers, and additional training requirements without providing reliable gains in precision. The deterministic keyword- and pattern-based semantic detector, while limited, provides auditable decisions and explicit negative evidence suppression, which proved critical for rejecting plot-heavy figures.

Preservation of rejected samples. Rather than discarding failures, rejected and uncertain figures are retained as debug artifacts. This decision was motivated by repeated observations that silent filtering masks systematic errors. Retaining rejected samples enables post-hoc analysis, threshold tuning, and defensible claims about dataset quality.

5.4 Dataset Quality Improvements Over Time

Compared to earlier pipeline variants, the final implementation exhibits measurable improvements:

Target n	Papers processed	Contributing papers	Zero-yield papers
50	51	15	36
100	153	35	118
150	268	63	205
200	387	93	294
250	477	114	363

Table 1: *Progressive yield as the pipeline is run until target size n is reached. "Papers processed" counts papers with a filled checkpoint entry. "Contributing papers" have ≥ 1 accepted circuit; "Zero-yield papers" were processed but produced 0 accepted circuits.*

- substantial reduction in partial and text-contaminated figure crops,
- significantly lower false-positive rate for plots and experimental figures,
- improved consistency across repeated runs due to deterministic processing,
- explicit traceability from dataset entries back to source papers and rejection decisions.

These improvements are not incidental; they are the direct result of replacing brittle heuristics and single-stage filters with a modular, multi-stage pipeline informed by observed failure cases

5.5 Quantitative Performance Analysis

The quantitative evaluation focuses on the pipeline's ability to selectively extract quantum circuit diagrams from a large corpus of scientific PDFs under realistic conditions. Rather than reporting classification accuracy against a labeled benchmark, performance is measured in terms of extraction yield, filtering effectiveness, and dataset sparsity handling.

Out of 3,553 extracted figure candidates, only 250 were retained as valid quantum circuits, illustrating the extreme sparsity of relevant figures in the *quant-ph* category. This corresponds to an acceptance rate of approximately 7%, highlighting the necessity of conservative filtering to avoid dataset pollution. Importantly, the pipeline consistently produces the same results across repeated runs, confirming deterministic behavior and reproducibility. These results demonstrate that the proposed approach scales to hundreds of papers while maintaining high precision in circuit selection, which is critical for downstream image-to-text dataset quality.

5.6 Scalability and Reproducibility

The pipeline is fully automated and relies on fixed paper ordering, explicit rate limiting during acquisition, and deterministic processing logic. Under the operational constraints specified by the project, the system can be executed with a configurable target size n, continuing sequential traversal of the *quant-ph* paper list until n validated quantum circuit images have been accepted (or the input list is exhausted), making the approach feasible for constructing datasets of arbitrary size subject to corpus sparsity and runtime constraints. Final outputs are aggregated into a structured JSON dataset stored in output/dataset_37_json/, accompanied by a paper-level summary file (output/paper_list_counts_37.csv) that records extraction counts in deterministic order. Because each stage is deterministic given the same code, configuration, and input list, repeated executions produce identical outputs, and the preserved provenance trail supports reproducible dataset construction despite PDF layout variability.

5.7 Limitations and Acceptability for the Task

The dataset remains incomplete and imperfect because it is produced fully automatically from various scientific PDFs. In this submission, the pipeline processed the full 26,908-paper candidate list, but yielded a relatively small accepted dataset: 114 papers contributed accepted circuits, producing 250 final circuit images. In total, 477 figure candidates were examined by the downstream filtering stages (including both accepted and rejected candidates). This extreme sparsity implies that recall cannot be guaranteed: many papers contain no circuits, and valid circuits may still be missed due to conservative thresholds, unusual drawing conventions, low-quality renders, or pages skipped under high vector-complexity constraints. Conversely, false positives cannot be eliminated entirely, because many non-circuit schematics (e.g., block diagrams, control-flow diagrams, and architecture figures) can resemble circuit layouts; such candidates occur frequently across the broader corpus and occasionally survive early filtering before being rejected or, in rare cases, incorrectly accepted without manual validation.

The dataset is constructed fully automatically from various scientific PDFs, and the *quant-ph* category serves as a broad search space rather than a circuit-only corpus. In this submission, the pipeline deterministically traversed the first **477 papers** in the fixed input list and stopped once the target size of **250 accepted quantum circuit images** was reached. Within these 477 processed papers, **114 papers** contributed at least one accepted circuit image, while **363 papers** produced zero accepted circuits. This outcome is expected because not all *quant-ph* publications are circuit-based quantum computing papers, and many papers contain no circuit figures. The pipeline's conservative extraction and filtering strategy prioritizes high-precision circuit inclusion and preserves a clear provenance trail (paper ID, page number, figure region) for every accepted sample, supporting reproducible dataset construction and systematic inspection of extraction outcomes.

6 Future Work

1. **Stronger hybrid filtering (vision + text).** Improve acceptance decisions by calibrating the CNN threshold and refining the semantic verifier with additional negative-context rules, with the option to prioritize borderline cases for minimal manual review.

2. **More robust figure extraction and deduplication.** Improve vector-figure reconstruction (better clustering of primitives, stronger multi-panel handling, adaptive rendering resolution) and reduce duplicate candidates via stronger cross-path deduplication.

3. **Richer metadata and scalable execution.** Improve caption-to-figure alignment and gate normalization (including parameter/notation variants), strengthen algorithm/protocol identification using lightweight ontologies or weak supervision, and scale runtime via parallel processing with caching/checkpointing while preserving determinism and provenance.

7 Conclusion

This project demonstrates a complete, fully automated pipeline for constructing a quantum circuit image-to-text dataset directly from scientific PDFs. Starting from a fixed *quant-ph* arXiv identifier list, the system deterministically acquires PDFs under explicit rate limiting, extracts figure candidates from both raster and vector content, and applies conservative hybrid filtering based on visual structure and caption/context evidence. The pipeline preserves provenance (paper ID, page number, figure region) and intermediate artifacts (raw candidates and rejected samples), enabling transparent auditing and reproducible reruns.

In this submission, the pipeline reached the required target of **250 accepted circuit images** after traversing **477 papers** in deterministic order. These images originate from **114 papers** that contributed at least one accepted circuit, while the remaining processed papers yielded zero accepted circuits, reflecting that *quant-ph* is a broad research category and not all publications contain circuit figures. This result supports feasibility at larger scale: the pipeline can be extended to collect additional images simply by continuing the same deterministic processing, with scalability primarily governed by corpus sparsity and runtime rather than methodological limitations.

The resulting metadata is intentionally grounded in what the papers explicitly report. When algorithm/problem naming is absent or ambiguous, the pipeline leaves the field unspecified rather than assigning weak labels, and gate mentions are extracted from caption/context text (average **3.44** gate mentions per circuit) as lightweight semantic cues rather than full executable circuit reconstructions. Within this intended scope—image-to-text supervision enriched with interpretable semantic signals and strong provenance—the dataset provides a practical and auditable foundation for training and analysis, and the pipeline provides a reproducible method that can be rerun and extended incrementally.

References

[1] arXiv, *arXiv API Access*. [Online]. Available: https://info.arxiv.org/help/api/index.html. Accessed: 2025-12-22.

[2] PyMuPDF Documentation, *OCR Recipe*. [Online]. Available: https://pymupdf.readthedocs.io/en/latest/recipes-ocr.html. Accessed: 2025-12-22.

[3] dpbmanalysis, "Quantum Circuit Images Dataset," Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/dpbmanalysis/quantum-circuit-images. Accessed: Jan. 2025.

[4] CVIT, IIIT Hyderabad, "DocFigure Dataset Page," 2019. [Online]. Available: https://cvit.iiit.ac.in/usodi/Docfig.php. Accessed: Jan. 2025.

[5] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 11966–11976, doi: 10.1109/CVPR52688.2022.01167. [Online]. Available: https://doi.org/10.1109/CVPR52688.2022.01167.

[6] H. Bast and C. Korzen, "A Benchmark and Evaluation for Text Extraction from PDF," in *Proceedings of the 17th ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, 2017, pp. 99–108, doi: 10.1109/JCDL.2017.7991564. [Online]. Available: https://doi.org/10.1109/JCDL.2017.7991564.

[7] N. Meuschke, A. Jagdale, T. Spinde, J. Mitrović, and B. Gipp, "A Benchmark of PDF Information Extraction Tools using a Multi-Task and Multi-Domain Evaluation Framework for Academic Documents," *arXiv preprint arXiv:2303.09957*, 2023. [Online]. Available: https://arxiv.org/abs/2303.09957.