

## Der finale Akzeptanz-Test (Anweisung an die KI)

"Die Implementierung der CV-Härtung (Prio 1, 2, 3) ist abgeschlossen. Die Architektur steht. Führe jetzt die folgenden drei Validierungs-Testläufe durch und berichte über das Ergebnis."

### Testlauf 1: Der "Text-Unterbrechungs-Test"

- **Ziel:** Validieren, ob **Text-Removal (Prio 2)** und **Gap-Bridging (Prio 3)** korrekt zusammenarbeiten.
- **Testbild:** Ein P&ID, bei dem ein **Text-Label eine durchgehende Rohrleitung kreuzt**.
- **Erfolgs-Kriterium:** Im finalen Output (`*_debug_map.png`) muss die Polyline als **eine durchgehende Linie** über die Text-Position hinweg gezeichnet werden (nicht als zwei getrennte Striche).

### Testlauf 2: Der "Lücken-Test" (Adaptive Thresholds)

- **Ziel:** Validieren, ob **Adaptive Thresholds (Prio 1)** das Matching-Problem bei Lücken löst.
- **Testbild:** Ein P&ID (idealerweise hochauflösend), bei dem eine Rohrleitung **sichtbar knapp (z.B. 5-10 Pixel) vor** der Bounding-Box eines Symbols endet.
- **Erfolgs-Kriterium:** Im finalen Output (`*_results.json`) muss die **connection** (z.B. `P-101 -> V-101`) trotzdem **korrekt vorhanden** sein, weil das adaptive Matching die Lücke toleriert hat.

### Testlauf 3: Der "Parallele-Linien-Stresstest"

- **Ziel:** Die *Grenzen* der aktuellen CV-Implementierung verstehen (die Schwachstelle, die wir *bewusst noch nicht* gefixt haben).
- **Testbild:** Ein P&ID mit **sehr eng anliegenden, parallelen Linien** (z.B. Vor- und Rücklauf).
- **Erfolgs-Kriterium (Hypothese):** Die KI soll bestätigen, dass die Skeletonization (Phase 2e) diese Linien wahrscheinlich zu **einer einzigen Linie verschmolzen** hat.
  - **WICHTIG:** Das ist für uns *akzeptabel*, solange die *logischen* Verbindungen im JSON (die `from_id` und `to_id`) korrekt sind.