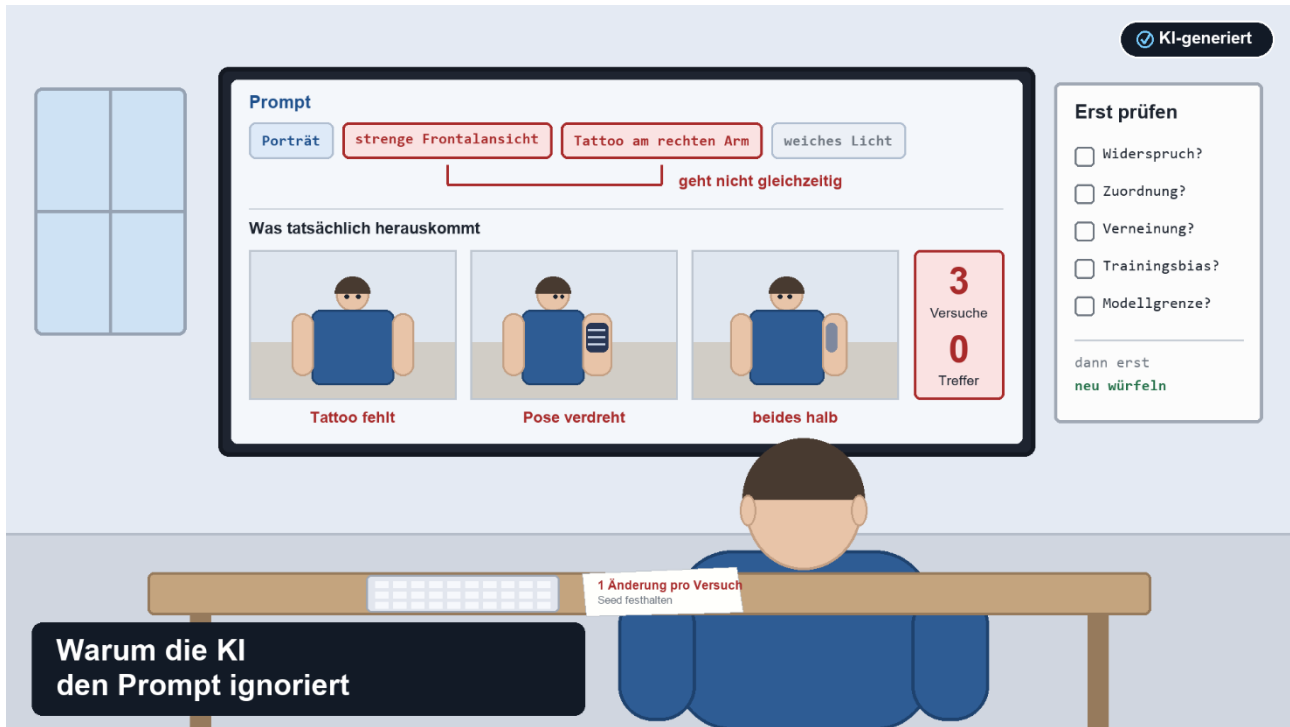


# Warum die KI den Prompt ignoriert



*Drei Versuche, kein Treffer: Der Prompt verlangt strenge Frontalansicht und ein gut sichtbares Tattoo am rechten Oberarm. Beides gleichzeitig gibt es nicht.*

| Wissen                                                                                             | Beratung                                                                                 | Schulung                                                                   |
|----------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------|----------------------------------------------------------------------------|
| Alle Grundlagen zur KI-Bildgenerierung an einer Stelle – vom Diffusionsprinzip bis zur Governance. | Wir bringen deine Bildproduktion von Rätselraten auf eine Trefferquote, die planbar ist. | Dein Team lernt, Fehlversuche zu diagnostizieren statt sie zu wiederholen. |
| <a href="#">KI-Bildgenerierung</a>                                                                 | <a href="#">KI-Bildgenerierung Beratung</a>                                              | <a href="#">KI-Bildgenerierung Schulung</a>                                |

Der Prompt war eindeutig. Da stand, was zu sehen sein soll, in ordentlichem Deutsch, mit Komma und allem. Und trotzdem liegt vor dir ein Bild, in dem genau das eine Detail fehlt, auf das es dir ankam. Also schreibst du den Prompt noch einmal. Diesmal länger. Dann mit Großbuchstaben. Dann mit drei Ausrufezeichen. Und irgendwann mit dem Wort „WICHTIG“ davor, weil man ja nie weiß.

Ich habe das bei einem Maschinenbauer erlebt, der ein einziges Porträt brauchte: eine Monteurin, streng frontal, Werkhalle im Hintergrund, und am rechten Oberarm ein Tattoo, das zur Bildsprache der Arbeitgeberkampagne gehörte. Die Marketingabteilung hatte über drei Tage knapp dreihundert Bilder erzeugt. In keinem war beides richtig. Entweder stand die Frau sauber frontal und der Arm war leer, oder das Tattoo war da und die Pose halb weggedreht. Die Diagnose dauerte dann keine zehn Minuten – der Prompt verlangte zwei Dinge, die sich physikalisch ausschließen. In einer streng frontalen Pose liegt die Außenseite des rechten Oberarms nun einmal nicht zur Kamera.

„KI ignoriert Prompt“ ist eine der häufigsten Suchanfragen rund um Bildgeneratoren, und sie führt fast immer zu derselben falschen Reaktion: lauter werden. Das Missverständnis dahinter ist, dass ein Bildmodell Anweisungen entgegennimmt und ausführt. Tut es nicht. Es nimmt deinen Text,

verwandelt ihn in Zahlen und sucht dann das Bild, das zu diesen Zahlen am besten passt. Ein Compiler würde dir einen Fehler melden, wenn dein Auftrag widersprüchlich ist. Ein Diffusionsmodell liefert stattdessen den Mittelweg – und der sieht meistens nach Ignoranz aus.

Die gute Nachricht: Die Ursachen lassen sich sortieren. Fünf davon sind dein Zuständigkeitsbereich und in Minuten behoben. Eine ist es nicht, und dort hört jede Prompt-Kunst auf. Der Unterschied zwischen beidem ist die eigentliche Fähigkeit, um die es in diesem Beitrag geht – denn wer ihn nicht erkennt, verbrennt Rechenzeit und Nerven an Problemen, die kein Prompt der Welt löst.

### Die Kurzfassung

Ein Bildmodell befolgt keine Anweisungen, sondern erzeugt das wahrscheinlichste Bild zu einer Zahlendarstellung deines Textes. Fünf Ursachen erklären fast jeden Fall von ignorierten Prompt-Teilen: ein Widerspruch im Prompt, falsch zugeordnete Merkmale (Prompt-Bleeding), das Verschmelzen ähnlicher Konzepte, Verneinungen, die der Text-Encoder nicht versteht, und Trainingsdaten-Bias, gegen den dein Wunsch anläuft. Dazu kommen echte Modellgrenzen: genaues Zählen, exakte räumliche Beziehungen, Hände und seltene Begriffe. Vier Tests – fünf Seeds, Prompt halbieren, Minimalprompt, Modellwechsel – trennen in wenigen Minuten die Prompt-Probleme von den Modellgrenzen.

## Warum ein Bildmodell keine Anweisungen befolgt

Bevor wir über Ursachen reden: Es hilft ungemein, sich klarzumachen, was zwischen deinem getippten Satz und dem fertigen Bild eigentlich passiert. Denn an jeder Station dieses Wegs kann etwas von deiner Absicht verloren gehen, und an jeder Station helfen andere Hebel. Wer alle Probleme mit einer besseren Formulierung lösen will, benutzt einen einzigen Schraubenschlüssel für vier verschiedene Schrauben.

### Dein Prompt wird zu einer Zahlenwolke

Der Text geht zuerst durch einen Text-Encoder. Der zerlegt ihn in Token und macht daraus einen Zahlenblock, die sogenannte Einbettung. Erst dieser Zahlenblock erreicht das eigentliche Bildmodell – dein Satz mit seiner Grammatik existiert ab hier nicht mehr. Welcher Encoder das ist, entscheidet, wie viel von deiner Formulierung überhaupt ankommt.

Die ältere Generation, also Stable Diffusion 1.5, SDXL und die große Familie an Community-Ablegern, arbeitet mit CLIP. CLIP wurde darauf trainiert, Bilder und kurze Bildunterschriften einander zuzuordnen. Es erkennt Konzepte, aber es liest keine Erzählung. Satzbau interessiert CLIP nicht: „die Katze sitzt links vom Hund“ und „Katze, Hund, links, sitzen“ landen in ähnlichen Zahlen. Dazu kommt, dass CLIP Text in Blöcken von 75 Token verarbeitet – längere Prompts werden zerschnitten, getrennt kodiert und wieder aneinandergehängt. Über solche Blockgrenzen hinweg fließen Merkmale besonders gern zum falschen Objekt.

Die neuere Generation hat den Encoder ausgetauscht und damit einen echten Sprung gemacht. FLUX.1 nutzt zusätzlich zu CLIP einen T5-Encoder. FLUX.2 [dev] arbeitet mit einem einzigen Text-Encoder auf Basis von Mistral Small 3.1 mit 24 Milliarden Parametern, die kleineren FLUX.2-[klein]-Varianten setzen auf Qwen3, Qwen-Image verwendet das Vision-Language-Modell Qwen2.5-VL, und Z-Image aus Alibabas Tongyi-Labor bringt seinen eigenen Sprachmodell-Encoder mit. Das sind ausgewachsene Sprachmodelle. Sie verstehen Nebensätze, Aufzählungen, Bedingungen und räumliche Angaben, und sie arbeiten mit einem zusammenhängenden Fenster von bis zu 512 Token statt mit aneinandergeklebten Blöcken. Wer heute eine Kommakette aus SD-1.5-Zeiten in FLUX.2 tippt, verschenkt genau die Fähigkeit, für die das Modell gebaut wurde.

Trotzdem gilt in beiden Welten: Der Encoder liefert eine Wolke aus Bedeutungen, keine Checkliste mit Haken. Welcher Modelltyp gerade vor dir liegt und was er mitbringt, klärt [Checkpoints, Modelle und VAE](#). Wie man für die jeweilige Familie überhaupt schreibt, steht ausführlich in [Prompt und Negativ-Prompt in der KI-Bildgenerierung](#).

## Aufmerksamkeit ist ein Budget, kein Befehl

Im Modell selbst entscheidet der Aufmerksamkeitsmechanismus, welcher Teil deines Textes welchen Teil der Bildfläche beeinflusst. Und hier steckt der Kern des ganzen Problems: Diese Zuordnung ist nicht fest verdrahtet, sondern das Ergebnis eines Wettbewerbs. Alle Begriffe deines Prompts konkurrieren gleichzeitig um dieselbe Fläche. Was sich durchsetzt, hängt davon ab, wie kräftig ein Begriff im Training vertreten war, wo er im Prompt steht und wie viel Konkurrenz er hat.

Daraus folgt eine Faustregel, die sich in jeder Testreihe bestätigt: Je länger dein Prompt, desto dünner die Aufmerksamkeit pro Begriff. Ein Prompt mit acht Bausteinen setzt jeden einzelnen davon zuverlässiger um als einer mit dreißig. Das ist der Grund, warum die übliche Reaktion – noch mehr hineinschreiben – das Problem regelmäßig verschlimmert statt löst.

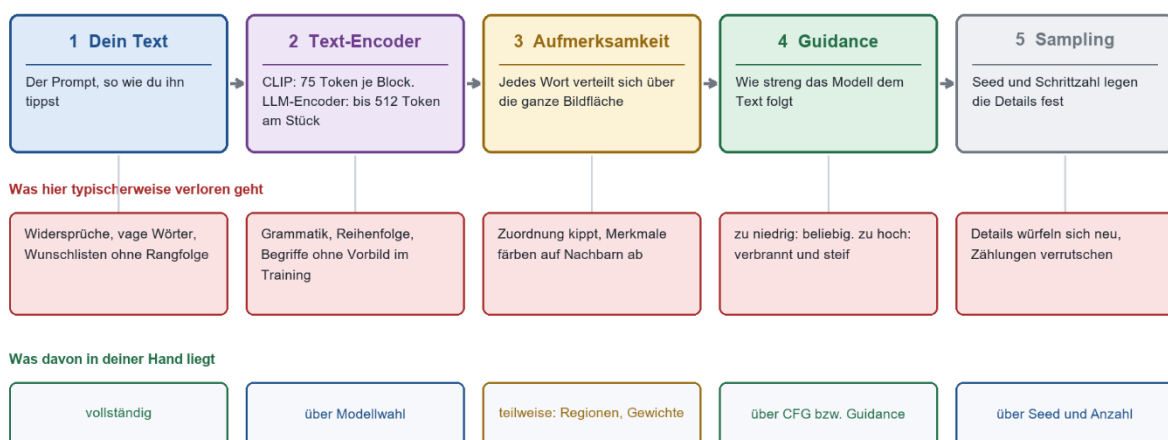
## Guidance regelt die Strenge, nicht das Verständnis

Der nächste Hebel ist die Guidance, bei klassischen Modellen als CFG-Scale bezeichnet. Sie bestimmt, wie stark sich das Modell an den Text klammert. Zu niedrig, und das Ergebnis wird beliebig; zu hoch, und das Bild wird steif, überzeichnet und farbverbrannt. Was Guidance nicht kann: aus einem unverstandenen Begriff einen verstandenen machen. Wer ein ignoriertes Detail über die CFG erzwingen will, dreht an der Lautstärke, nicht am Inhalt. Die Wirkungskurve und ihre Grenzen beschreibt [CFG-Scale: Guidance in KI-Bildgeneratoren](#).

Bei destillierten Modellen ist dieser Hebel ohnehin weitgehend blockiert. FLUX.1 [dev] und [schnell] arbeiten mit einem eingebauten Guidance-Wert statt mit echtem CFG, FLUX.2 [dev] ebenfalls, und Z-Image Turbo läuft in acht Schritten mit CFG-Werten um 1,5 bis 2,0. Wer dort die üblichen Werte zwischen 6 und 9 einträgt, bekommt kein gehorsameres Bild, sondern ein kaputtes. Und das Feld für den Negativ-Prompt ist in vielen Oberflächen zwar sichtbar, aber ohne einen zweiten Rechenzweig wirkungslos.

## Wo dein Prompt unterwegs verloren geht

Fünf Stationen zwischen deinem Text und dem fertigen Bild



Ein lauterer Prompt hilft nur an Station 1 – an allen anderen Stellen brauchst du andere Hebel.

*Fünf Stationen zwischen deinem Text und dem Bild. Nur an der ersten hilft eine andere Formulierung – an den übrigen brauchst du andere Werkzeuge.*

Am Ende steht das Sampling: Der Sampler baut aus Rauschen schrittweise das Bild, und der Seed legt fest, mit welchem Rauschen er beginnt. Zwei Seeds bei ansonsten identischen Einstellungen liefern zwei verschiedene Bilder – manchmal ist das strittige Detail beim einen korrekt und beim anderen nicht. Das ist keine Frage der Formulierung, sondern schlicht Zufall, und es ist die Erklärung dafür, dass derselbe Prompt mal funktioniert und mal nicht.

### Ignorieren ist kein Trotz, sondern Statistik

Das Modell verweigert nichts. Es hat kein Konzept von Anweisung, Priorität oder Wichtigkeit. Ausrufezeichen, Großbuchstaben und Formulierungen wie „unbedingt“ oder „auf keinen Fall“ sind für den Text-Encoder normale Token, die Aufmerksamkeit verbrauchen und nichts bewirken. Der einzige Weg, einem Baustein mehr Gewicht zu geben, führt über seine Position im Prompt und über die Gewichtungssyntax deiner Oberfläche – nicht über Nachdruck.

## Die fünf Ursachen, sortiert nach Häufigkeit

In der Praxis lassen sich fast alle Fälle auf fünf Ursachen zurückführen. Sie stehen hier in der Reihenfolge, in der du prüfen solltest – nicht in der Reihenfolge, in der sie spannend sind. Die häufigste Ursache ist gleichzeitig die, die man an sich selbst am schlechtesten sieht.

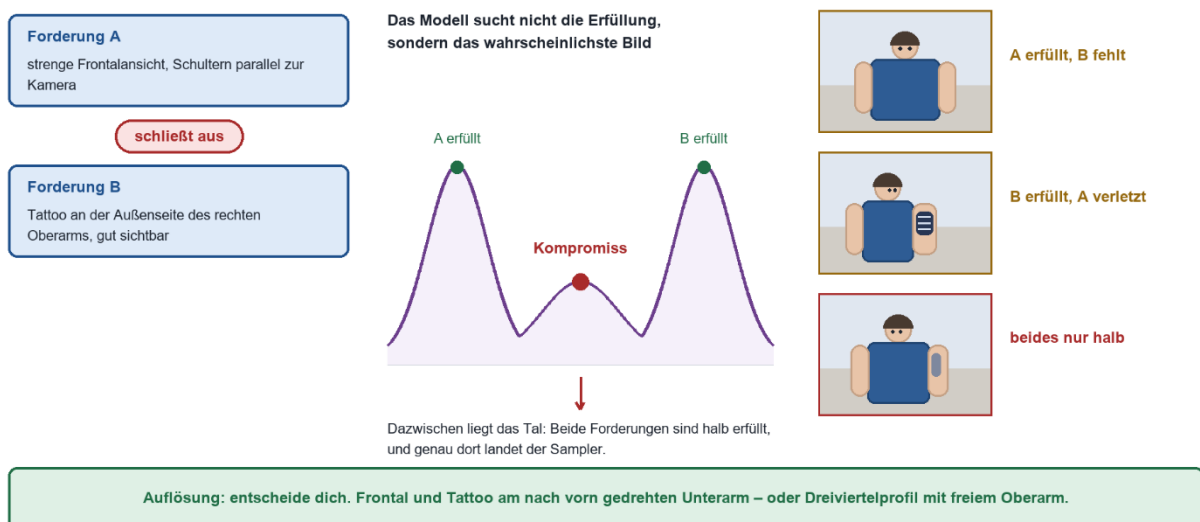
### Ursache 1: Der Prompt widerspricht sich selbst

Zwei Anforderungen, die sich gegenseitig ausschließen, sind der mit Abstand häufigste Grund. Das Tückische daran: Ein Widerspruch fühlt sich beim Tippen nicht wie einer an. Du schreibst „strenge Frontalansicht“ und ein paar Wörter später „Tattoo an der Außenseite des rechten Oberarms, gut sichtbar“ – beides ist für sich genommen vernünftig. Nur eben nicht gleichzeitig.

Das Modell kann nicht zurückfragen. Es sucht in seiner Wahrscheinlichkeitslandschaft nach dem Bild, das zu beiden Textteilen halbwegs passt, und landet im Tal zwischen zwei Gipfeln: eine leicht verdrehte Pose, der Arm angeschnitten, das Tattoo halb da. Kein Teil der Anforderung ist erfüllt, aber jeder ist ein bisschen berücksichtigt. Genau dieser Kompromiss ist es, der wie Ignoranz aussieht.

### Der unsichtbare Widerspruch

Zwei Wünsche, die nicht gleichzeitig gelten können



*Das Leitbeispiel: Zwei erfüllbare Gipfel, dazwischen ein Tal. Der Sampler landet im Tal, weil dort beide Textteile ein wenig zutreffen.*

Typische versteckte Widersprüche, die mir immer wieder begegnen: Weitwinkel mit starker Hintergrundunschärfe. Eine menschenleere Halle, in der ein Team arbeiten soll. Blaue Stunde mit hartem Sonnenlicht. Vogelperspektive mit Blickkontakt in die Kamera. Saubere Studiofreistellung mit natürlichem Fensterlicht. Ein Logo auf einem Kleidungsstück, das im Faltenwurf verschwindet. Jedes dieser Paare erzeugt denselben unbrauchbaren Mittelweg.

### **Der Fotografen-Test**

Lies deinen Prompt und frage dich: Könnte ein Mensch mit einer Kamera dieses Bild in einer einzigen Aufnahme machen? Wenn die Antwort nein lautet, wirst du es auch von einem Bildmodell nicht bekommen – jedenfalls nicht in einem Durchgang. Bei dem Tattoo-Beispiel lautet die Lösung entweder Dreiviertelprofil mit freiem Oberarm oder Frontalansicht mit angewinkeltem Arm, sodass die tätowierte Seite zur Kamera zeigt. Beides steht dann ausdrücklich im Prompt, und der Widerspruch ist weg.

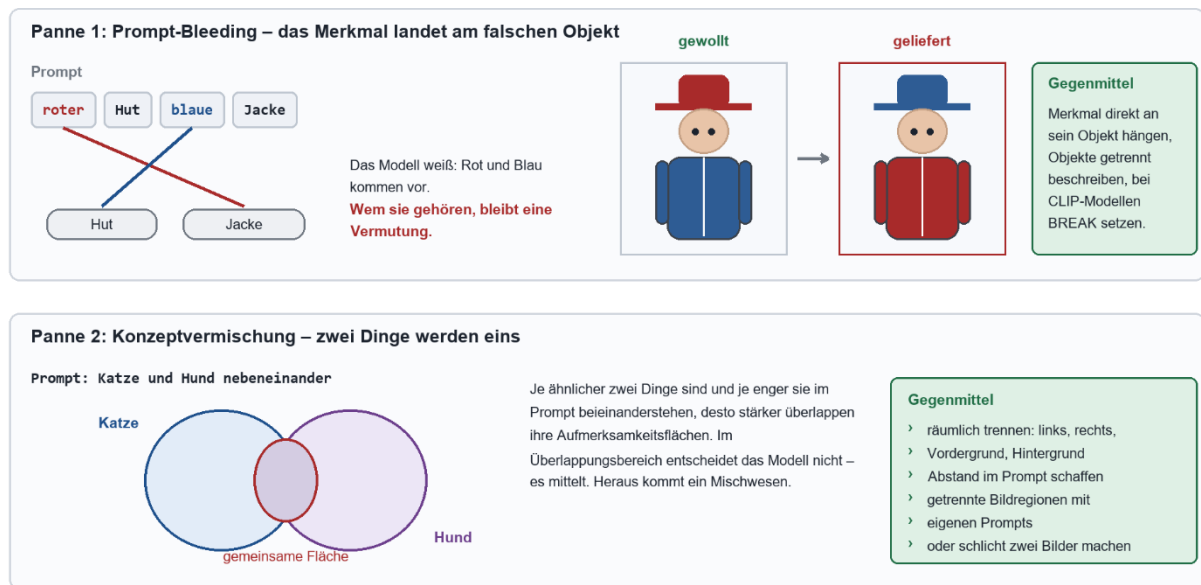
## **Ursache 2: Prompt-Bleeding – das Merkmal landet am falschen Objekt**

Du schreibst „roter Hut, blaue Jacke“ und bekommst einen blauen Hut über einer roten Jacke. Das Modell hat verstanden, dass Rot und Blau im Bild vorkommen sollen, und auch, dass es einen Hut und eine Jacke gibt. Nur die Zuordnung ist ihm entglitten. In der Forschung heißt das Attribut-Bindung, und es ist kein Randproblem: Ganze Verfahrensfamilien wie Attend-and-Excite oder die späteren Ansätze zur Ausrichtung von Aufmerksamkeitskarten sind ausschließlich dafür entwickelt worden, diese Zuordnung zu stabilisieren.

Der Grund liegt in der Art, wie der Text kodiert wird. Ein Farbwort färbt bereits im Encoder auf benachbarte Begriffe ab – deshalb der englische Name Prompt-Bleeding oder Concept Bleeding. Bei CLIP-Modellen ist der Effekt am stärksten, besonders über die Grenzen der 75-Token-Blöcke hinweg. Modelle mit Sprachmodell-Encoder sind deutlich besser, weil sie den Satzbau tatsächlich auswerten. Verschwunden ist das Problem aber auch dort nicht, sobald mehrere gleichartige Objekte im Bild sind.

## Wenn Merkmale wandern und Konzepte verschmelzen

Prompt-Bleeding und Konzeptvermischung – zwei verwandte Pannen



*Zwei verwandte Pannen: Das Merkmal wandert zum falschen Objekt, und zwei ähnliche Konzepte teilen sich eine Fläche, bis nur noch eines übrig ist.*

### Ursache 3: Ähnliche Konzepte verschmelzen

Der nächste Schritt derselben Mechanik: Zwei Dinge werden nicht vertauscht, sondern zu einem einzigen verrührt. „Eine Katze und ein Hund nebeneinander“ liefert mit erstaunlicher Zuverlässigkeit ein Mischwesen mit Katzenkopf und Hundekörper. Dasselbe passiert bei zwei Personen in ähnlicher Arbeitskleidung, bei zwei Produkten derselben Familie oder bei einem Logo, das neben einem anderen Logo stehen soll.

Die Ursache ist die geteilte Aufmerksamkeitsfläche. Solange das Modell nicht eindeutig weiß, wo das eine aufhört und das andere anfängt, mittelt es im Überlappungsbereich. Was hilft, ist räumliche Eindeutigkeit: „links im Bild ... rechts im Bild ...“, deutlich verschiedene Beschreibungen für beide Objekte, und im Zweifel zwei Bilder statt eines. Wer es präzise braucht, arbeitet mit getrennten Bildregionen, die jeweils ihren eigenen Prompt bekommen – in ComfyUI ist das unter dem Stichwort Attention Couple oder regionales Prompting ein Standardaufbau: eine Maske je Bereich, ein Prompt je Maske, alles zusammen in einen Sampler.

### Ursache 4: Verneinungen laufen ins Leere

„Ein aufgeräumter Schreibtisch, keine Kaffeetasse“ – und im Bild steht eine Kaffeetasse. Das ist kein Zufall und keine Bosheit. Das Wort „Kaffeetasse“ steht in deinem Prompt, und der Encoder macht daraus das, was er immer macht: ein Konzept namens Kaffeetasse. Die Verneinung davor ist für ihn kaum mehr als ein Füllwort.

Das ist inzwischen sauber belegt. Untersuchungen zum Umgang von Vision-Language-Modellen mit Verneinung zeigen, dass Modelle dieser Bauart bei Aufgaben mit „kein“ oder „ohne“ nahe an Zufallsniveau liegen. Es gibt Gegenmaßnahmen aus der Forschung – Encoder, die gezielt auf Verneinungen nachtrainiert wurden, oder Verfahren, die den verneinten Anteil rechnerisch aus der Einbettung herausziehen. In den Modellen, die du im Alltag benutzt, steckt davon bislang wenig. Für dich heißt das: positiv formulieren. Nicht „keine Kaffeetasse“, sondern „eine leere, glatte Tischplatte“.

Der zweite saubere Weg ist der Negativ-Prompt – aber nur dort, wo er technisch wirkt, nämlich bei Modellen mit echtem CFG über 1. Bei destillierten Modellen ist das Feld Dekoration. Welche Familie was unterstützt und welche Ersatzverfahren es gibt, steht in [Prompt und Negativ-Prompt in der KI-Bildgenerierung](#).

### Die Negativliste, die niemand liest

Über Jahre gewachsene Negativlisten mit vierzig Begriffen sind in vielen Teams Standardausstattung. An SD 1.5 und SDXL wirken sie tatsächlich. An FLUX.1, FLUX.2 [dev] und anderen destillierten Modellen wirken sie nicht – dort gibt es keinen zweiten Rechenzweig, in den sie eingespeist werden könnten. Bevor du die Liste weiter pflegst, prüfe mit zwei Testbildern, ob sie überhaupt etwas verändert: einmal mit, einmal ohne, gleicher Seed. Wenn beide Bilder identisch sind, hast du soeben eine Arbeitsroutine eingespart.

## Ursache 5: Der Trainingsdaten-Bias zieht dagegen

Manchmal ist der Prompt widerspruchsfrei, die Zuordnung stimmt, Verneinungen sind keine drin – und trotzdem bekommst du etwas anderes, als du beschrieben hast. Dann läufst du gegen die Statistik der Trainingsdaten. Das Modell hat eine sehr konkrete Vorstellung davon, wie ein „Büro“, eine „Werkstatt“ oder eine „Führungskraft“ aussieht, und diese Vorstellung stammt aus Millionen Bildern, die überwiegend nicht aus deinem Land, deiner Branche und deinem Jahrzehnt stammen.

Das äußert sich in Kleinigkeiten mit großer Wirkung: Aus der deutschen Innenstadt wird eine amerikanische Straßenszene mit Ampeln an Auslegern. Die Werkbank ist aufgeräumt wie im Prospekt, obwohl du „gebrauchte Werkstatt“ geschrieben hast. Alle Personen lächeln in die Kamera. Die Handwerkerin wird zum Handwerker, weil die Trainingsbilder zu diesem Beruf überwiegend Männer zeigen. Der Prompt wird nicht ignoriert – er wird überstimmt.

Dagegen hilft zuerst Konkretheit: Je genauer du das beschreibst, was von der Erwartung abweicht, desto eher setzt es sich durch. Reicht das nicht, brauchst du Material statt Worte – ein Referenzbild in einem Modell, das Bildeingaben verarbeitet, oder ein trainierter Zusatz für deine Produkte, deine Leute und deine Umgebung. Wie das funktioniert und was es kostet, erklärt [Was ist ein LoRA?](#). Wie man daraus eine Bildsprache macht, die über Monate trägt, beschreibe ich in [Markenkonsistente KI-generierte Bilder](#).

| Symptom                                                    | Wahrscheinliche Ursache                      | Wirksamer Hebel                                           |
|------------------------------------------------------------|----------------------------------------------|-----------------------------------------------------------|
| Ein Detail fehlt dauerhaft, egal wie oft du es wiederholst | Widerspruch zu einer anderen Anforderung     | widersprüchlichen Baustein streichen oder Pose ändern     |
| Farben sitzen am falschen Objekt                           | Prompt-Bleeding bei der Attribut-Bindung     | Merkmal direkt neben sein Objekt, bei CLIP-Modellen BREAK |
| Zwei Objekte werden zu einem                               | geteilte Aufmerksamkeitsfläche               | räumlich trennen, Regionen setzen, notfalls zwei Bilder   |
| Das Ausgeschlossene erscheint erst recht                   | Verneinung im positiven Prompt               | positiv umformulieren oder Negativ-Prompt nutzen          |
| Das Bild wirkt generisch statt nach deinem Betrieb         | Trainingsdaten-Bias                          | konkreter beschreiben, Referenzbild, LoRA                 |
| Der letzte Satz des Prompts kommt nie an                   | Aufmerksamkeit verteilt sich, Prompt zu lang | vorziehen statt gewichten, Prompt kürzen                  |
| Die Anzahl stimmt nie                                      | bekannte Modellschwäche beim Zählen          | kleine Mengen, Komposition vorgeben, nachbearbeiten       |
| Ergebnis schwankt stark bei gleichem Prompt                | Seed-Zufall                                  | Seed festhalten, mehr Varianten, Treffer sichern          |

## Am Prompt feilen oder gegen eine Modellgrenze rennen

Die fünf Ursachen aus dem letzten Kapitel haben eines gemeinsam: Sie lassen sich lösen. Es gibt aber eine sechste Kategorie, bei der jede weitere Runde Prompt-Arbeit verlorene Zeit ist – nämlich dann, wenn das Modell das Gewünschte schlicht nicht kann. Diese Unterscheidung ist die wichtigste Fähigkeit im ganzen Thema, und sie ist erlernbar.

### Was Modelle strukturell schlecht können

Die Schwächen sind seit Jahren dieselben, und sie sind gut vermessen. Der Prüfstand GenEval bewertet Text-Bild-Treue in einzelnen Disziplinen, und das Muster wiederholt sich über alle Modellgenerationen: Objekte erzeugen und ihnen Eigenschaften geben klappt zuverlässig. Genaues Zählen, exakte räumliche Beziehungen und Handlungen zwischen mehreren Objekten sind die Dauerbaustellen. „Drei Äpfel“ werden zu zwei oder vier, „links vom Baum“ landet rechts, und „die Frau reicht dem Mann den Schlüssel“ ergibt zwei Personen und irgendwo einen Schlüssel.

Der Grund ist derselbe wie bei den anderen Pannen: Zahlwörter und Ortsbestimmungen sind für den Aufmerksamkeitsmechanismus schwache Signale. Sie beschreiben keine sichtbare Eigenschaft, sondern eine Beziehung – und Beziehungen haben im Training viel seltener ein eindeutiges Bildkorrelat als Farben oder Materialien. Bei Modellen mit Sprachmodell-Encoder ist es besser geworden, gerade bei einfachen Angaben wie links und rechts. Auf eine exakte Zahl über etwa fünf Objekte würde ich trotzdem kein Projekt aufbauen.

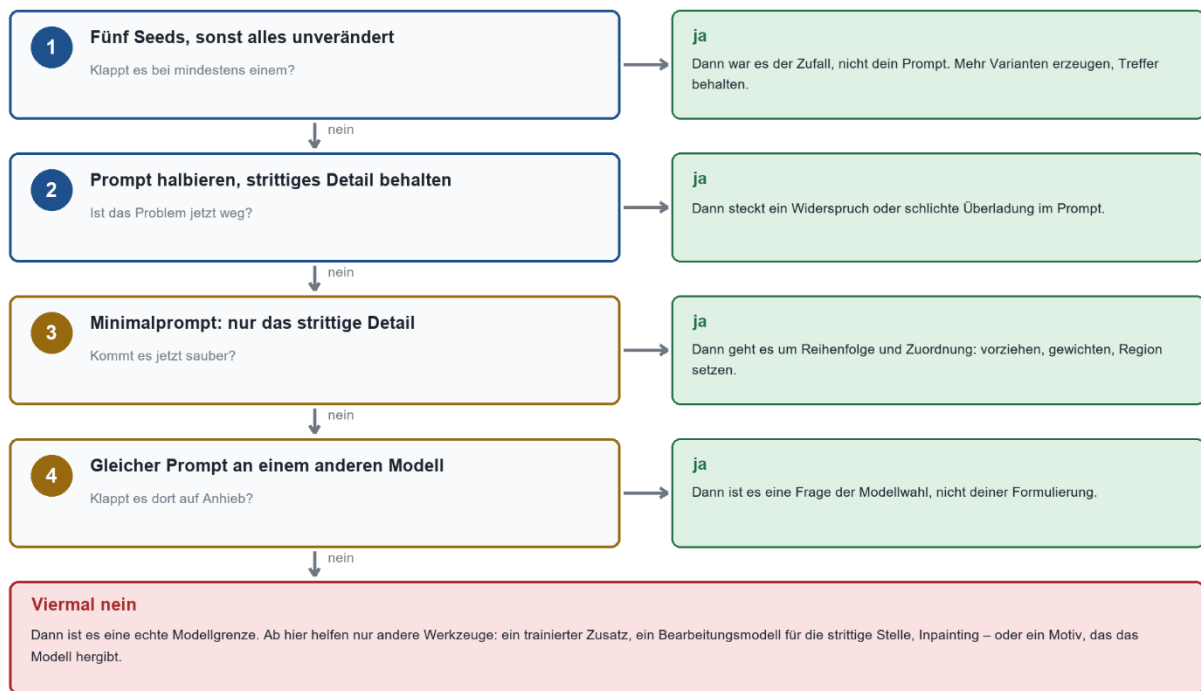
Dazu kommen die klassischen Detailschwächen. Hände und Finger sind der prominenteste Fall, und sie sind ein eigenes Kapitel wert – warum ausgerechnet diese Körperteile so hartnäckig sind, steht in [Warum KI keine Hände kann](#). Ähnlich gelagert: Spiegelungen, die physikalisch stimmen sollen, Schriftzüge in exakter Schreibweise, Markenlogos mit korrekten Proportionen und Nischenbegriffe, die im Training kaum vorkamen. Ein Modell, das ein Bauteil nicht kennt, erfindet etwas, das so ähnlich aussieht. Kein Prompt der Welt ändert daran etwas.

### Vier Tests, die die Frage in Minuten klären

Statt zu raten, testest du. Die folgende Reihenfolge hat sich bewährt, weil jeder Schritt eine andere Hypothese ausschließt und keiner länger als ein paar Minuten dauert. Wichtig ist nur: Du änderst pro Test genau eine Sache. Wer gleichzeitig Prompt, Sampler und Schrittzahl verstellt, produziert Bilder, aber keine Erkenntnis.

## Am Prompt feilen oder gegen eine Modellgrenze rennen?

Vier Tests in fester Reihenfolge — jeder dauert wenige Minuten



*Vier Tests in fester Reihenfolge. Wer sie durchläuft, weiß nach zwanzig Minuten, woran er ist – statt nach zweihundert Bildern immer noch zu raten.*

Test eins ist der billigste: fünf verschiedene Seeds, alles andere unverändert. Kommt das strittige Detail bei einem davon korrekt heraus, war es nie ein Prompt-Problem, sondern die Zufallskomponente. Dann brauchst du keine bessere Formulierung, sondern mehr Versuche – und ein Verfahren, das die Treffer sichert. Wie Seeds funktionieren und warum sie beim Testen festgehalten gehören, steht in [Der Seed bei der KI-Bildgenerierung](#). Wie viele Varianten pro Durchlauf sinnvoll sind, klärt [batch size und Reproduzierbarkeit](#).

Test zwei: Halbiere den Prompt und behalte das strittige Detail. Ist das Problem weg, war es Überladung oder ein Widerspruch – dann fügst du die gestrichenen Teile einzeln wieder hinzu, bis der Fehler zurückkommt. Der Baustein, bei dem das passiert, ist dein Übeltäter. Das dauert vier Durchläufe und ersetzt vierzig Rateversuche.

Test drei: ein Minimalprompt, der nur noch aus dem strittigen Detail besteht. „Nahaufnahme eines tätowierten rechten Oberarms“ – kommt das sauber, kann das Modell es grundsätzlich, und du hast ein Zuordnungs- oder Reihenfolgeproblem. Kommt es auch allein nicht, ist der Verdacht auf eine echte Grenze stark.

Test vier: derselbe Prompt an einem anderen Modell, idealerweise aus einer anderen Familie. Klappt es dort auf Anhieb, ist es eine Frage der Modellwahl. Das ist häufiger, als viele denken – die Familien unterscheiden sich in ihren Stärken erheblich. Eine Einordnung liefert [Die 15 wichtigsten Modelle zur lokalen Bildgenerierung](#); wer noch beim Auswählen ist, findet in der [Übersicht der wichtigsten KI-Bildgeneratoren](#) die Landkarte dazu.

| Gehört dir                     | Gehört dem Modell                         |
|--------------------------------|-------------------------------------------|
| Widersprüche im Prompt         | Begriffe, die im Training fehlen          |
| Reihenfolge und Länge          | genaues Zählen über wenige Objekte hinaus |
| Verneinungen und Negativlisten | exakte räumliche Beziehungen              |

|                                             |                                           |
|---------------------------------------------|-------------------------------------------|
| Zuordnung von Merkmalen zu Objekten         | Hände, Finger, komplexe Anatomie          |
| Wahl der Schreibweise für die Modellfamilie | physikalisch korrekte Spiegelungen        |
| Gewichtung und Regionen                     | fehlerfreie Schrift in beliebiger Sprache |
| Wahl des Modells                            | Stilgrenzen des trainierten Materials     |

### Setz dir ein Zeitbudget

Zwanzig Minuten Diagnose sind fast immer billiger als zweihundert weitere Bilder. Leg vor dem Start fest, wie viele Durchläufe ein Motiv bekommt – bei uns sind es zwölf. Wer danach keinen brauchbaren Treffer hat, wechselt das Werkzeug statt die Wortwahl: anderes Modell, Referenzbild, Nachbearbeitung der strittigen Stelle oder ein abgewandeltes Motiv. Das ist keine Kapitulation, sondern Projektarbeit.

## Was tatsächlich hilft

Die Werkzeugkiste ist größer, als die meisten nutzen. Sie reicht von Umformulierung über Gewichtung und Regionen bis zu Verfahren, die gar nicht mehr mit Sprache arbeiten. Die Reihenfolge hier ist auch die Reihenfolge des Aufwands: oben kostenlos, unten mit Vorbereitung verbunden.

### Sagen, was da sein soll

Die wirksamste Einzelmaßnahme ist gleichzeitig die einfachste: Beschreibe Anwesenheit statt Abwesenheit, und beschreibe konkret statt schmückend. „Schönes Licht“, „hochwertig“, „professionell“ und „modern“ sind über Millionen völlig verschiedener Bilder verteilt und bedeuten für das Modell fast nichts – sie verbrauchen nur Aufmerksamkeit. „Weiches Fensterlicht von links, weiche Schatten nach rechts“ ist dagegen eine Ansage, die im Trainingsmaterial eindeutig belegt ist.

### Vorziehen schlägt gewichten

Was früh im Prompt steht, bekommt in allen gängigen Encodern mehr Gewicht. Bevor du also mit Klammern arbeitest, verschiebe das strittige Detail nach vorn. Erst wenn das nicht reicht, kommt die Gewichtungssyntax: (rote Jacke:1.3) betont, (rote Jacke:0.7) nimmt zurück. Der Arbeitsbereich liegt zwischen 1,1 und 1,4; darüber wird es schnell hässlich, weil das Modell nie auf derart aufgeblasene Einbettungen trainiert wurde. Und Vorsicht beim Übernehmen fremder Prompts: Oberflächen der Automatic1111-Familie normalisieren nach dem Gewichten, ComfyUI multipliziert in der Voreinstellung direkt – dieselbe Zahl wirkt dort spürbar härter.

### Trennen statt drängeln

Wenn zwei Dinge sich ständig ins Gehege kommen, hilft Trennung mehr als jede Formulierung. Die einfachste Form ist die räumliche Ansage im Prompt, die bei Modellen mit Sprachmodell-Encoder gut ankommt. Die harte Form sind getrennte Bildregionen mit je eigenem Prompt, wie sie regionales Prompting in ComfyUI erlaubt: eine Maske je Bereich, ein Prompt je Maske. Und die ehrlichste Form ist, zwei Bilder zu erzeugen und sie zusammenzusetzen – was in der Praxis oft schneller geht als der fünfzigste Versuch, das Modell zur Vernunft zu bringen.

### Wenn Sprache nicht reicht, nimm Bilder

Der größte Fortschritt der letzten beiden Jahre liegt nicht in besseren Prompts, sondern darin, dass Sprache nicht mehr der einzige Eingabekanal ist. Ein Bearbeitungsmodell nimmt dein fertiges Bild plus eine Anweisung entgegen und ändert gezielt eine Stelle, statt alles neu zu würfeln. Qwen-Image-Edit aus dem Tongyi-Labor arbeitet genau so, FLUX.2 verarbeitet Referenzbilder direkt im Prompt-Kontext. Für einzelne Stellen bleibt Inpainting das Mittel der Wahl: Bereich markieren,

nur diesen Bereich neu erzeugen, Rest unangetastet. Für Posen und Bildaufbau gibt es Steuerungsmodelle, die eine Skizze oder ein Posengerüst als Vorgabe nehmen.

Und wenn immer wieder dasselbe fehlt – dein Produkt, dein Gesicht, dein Werkstattstil –, ist ein trainierter Zusatz die saubere Lösung statt eines immer längeren Prompts. Was ein LoRA leistet, was das Training kostet und wann es sich rechnet, steht in [Was ist ein LoRA?](#).

| Maßnahme                             | Hilft bei                             | Aufwand             | Risiko                                 |
|--------------------------------------|---------------------------------------|---------------------|----------------------------------------|
| Positiv umformulieren                | Verneinungen, vage Begriffe           | Minuten             | keins                                  |
| Baustein vorziehen                   | ignorierte Details am Prompt-Ende     | Minuten             | keins                                  |
| Widersprüchlichen Baustein streichen | Kompromissbilder                      | Minuten             | Motiv ändert sich                      |
| Gewichtung 1,1 bis 1,4               | zu schwache oder zu starke Details    | Minuten             | Artefakte ab 1,6                       |
| Negativ-Prompt                       | Störer bei Modellen mit CFG über 1    | Minuten             | wirkungslos bei destillierten Modellen |
| Regionen mit eigenen Prompts         | Verschmelzen, mehrere Motive          | Stunden             | Übergänge müssen sitzen                |
| Inpainting                           | eine einzelne kaputte Stelle          | Minuten bis Stunden | sichtbare Nähte                        |
| Bearbeitungsmodell mit Anweisung     | gezielte Änderung am fertigen Bild    | Stunden             | Stilbruch zum Original                 |
| Anderes Modell                       | familiärentypische Schwächen          | Stunden             | Prompt muss neu gebaut werden          |
| LoRA trainieren                      | wiederkehrende Motive, Stil, Personen | Tage                | Pflegeaufwand, Datenrechte             |

### Die Industrie der magischen Wörter

Es gibt einen erstaunlich großen Markt für „perfekte Prompt-Formeln“, und das meiste davon verspricht mehr Kontrolle, als technisch vorhanden ist. Wortketten wie „masterpiece, best quality, 8k, award winning“ stammen aus SD-1.5-Zeiten, wo sie tatsächlich auf Bewertungstexte in den Trainingsdaten zielten. An modernen Modellen verbrauchen sie Token und bewirken bestenfalls nichts. Ein guter Prompt holt das Beste aus einem Modell heraus – er hebt dessen Grenzen nicht auf.

## Im Unternehmen: aus Frust wird ein Verfahren

Alles bisher Gesagte gilt für eine Person an einem Rechner. Sobald mehrere Leute Bilder erzeugen, kommt ein zweites Problem dazu: Niemand weiß mehr, welcher Prompt zu welchem Bild gehörte und was beim letzten Mal geholfen hat. Dann wird dieselbe Diagnose alle drei Wochen neu gestellt – von jeder Person einzeln, mit jeweils zweihundert Bildern.

### Ein Parameter pro Versuch, und ein Protokoll

Die Grundregel ist unspektakulär und wird trotzdem ständig gebrochen: eine Änderung pro Versuch, Seed festhalten, alles andere unverändert lassen. Sampler, Schrittzahl und Guidance gehören für die Dauer einer Testreihe eingefroren – wie diese Regler wirken, steht in [Sampler & Scheduler erklärt](#) und [Sampling-Steps erklärt](#). Und dokumentiert wird nicht der Erfolg, sondern der Versuch: Was war die Hypothese, was wurde geändert, was kam heraus.

Genau dafür setze ich im eigenen Betrieb UB.Dossier ein: Prompt-Bausteine, Modellstände und Bildkonfigurationen liegen dort als gepflegter Katalog mit Nachweis, nicht als gewachsene Textdatei auf irgendeinem Laufwerk. Den Produktionsteil – also den Ablauf selbst, seine Änderungen und den Plausibilitäts-Check vor dem Durchlauf – übernimmt UB.Forge. Wie das

zusammenspielt, beschreibe ich in [Dogfooding: Ich setze ein, was ich berate](#). Wer die Ablage lieber auf vorhandener Microsoft-Infrastruktur aufbaut, findet den Weg in [Digital Asset Management für KI-Bilder mit SharePoint Online und Power Platform](#).

## Abnahmekriterien vor dem ersten Bild

Die meisten Endlosschleifen entstehen, weil niemand vorher festgelegt hat, was das Bild leisten muss. „Sieht gut aus“ ist kein Kriterium, sondern eine Meinung. Drei bis fünf harte Punkte reichen: Bildausschnitt, Blickrichtung, welche Details zwingend erkennbar sein müssen, welche Farbwelt gilt, wo Text hinein soll. Das dauert zehn Minuten und beendet die Diskussion darüber, ob das vierzigste Bild nun besser ist als das zwölfte.

Zum Verfahren gehört auch, was nach dem Bild kommt. Seit dem 2. August 2026 greifen die Transparenzpflichten des EU AI Act für KI-erzeugte Inhalte – was das konkret für Bilder im Marketing bedeutet, steht in [EU AI Act: Was seit dem 2. August 2026 gilt](#). Ob ein Bild überhaupt als KI-erzeugt erkennbar ist und woran, behandelt [KI-generierte Bilder erkennen](#).

## Wo die Bilder entstehen, ändert die Diagnose

Ein praktischer Nebeneffekt der Werkzeugwahl: In einer Cloud-Oberfläche kannst du oft weder den Seed festhalten noch das Modell wechseln noch Regionen setzen – damit fallen drei der vier Diagnosetests aus. Wer regelmäßig produziert, hat lokal schlicht mehr Hebel. Die Abwägung dazu liefert [Cloud vs. lokale KI-Bildgenerierung](#); die strategische Sicht darauf [KI-Bildgenerierung im Unternehmen – Die strategische Alternative zur Cloud](#). Und wenn aus Einzelbildern eine laufende Produktion wird, hilft [Aufbau einer skalierbaren Bild-Maschine mit ComfyUI und Flux](#) beim Aufbau.

Wenn du das nicht allein aufbauen willst: Genau diese Verfahren – Diagnose, Prompt-Katalog, Abnahme, Nachweis – richte ich mit Teams ein. Was dabei herauskommt und wie ein solches Projekt abläuft, steht auf der Seite zur [KI-Bildgenerierung Beratung](#). Wer sein Team lieber selbst befähigen möchte, findet das passende Format unter [KI-Bildgenerierung Schulung](#).

## Fazit

Die KI ignoriert deinen Prompt nicht. Sie versteht ihn anders, als du ihn gemeint hast – und wo sie ihn versteht, kann sie ihn manchmal trotzdem nicht umsetzen. Diese beiden Fälle auseinanderzuhalten ist der ganze Trick. Fünf Ursachen liegen in deiner Hand: Widersprüche, falsche Zuordnung, verschmelzende Konzepte, Verneinungen und ein Trainingsbias, gegen den du zu leise formulierst. Sie alle lassen sich in Minuten prüfen und meist ebenso schnell beheben.

Was nicht in deiner Hand liegt, sind die strukturellen Schwächen: genaues Zählen, exakte Positionen, Hände, fehlende Begriffe. Dort hilft kein Nachdruck, sondern ein anderes Werkzeug – ein anderes Modell, ein Referenzbild, ein trainierter Zusatz, eine gezielte Nachbearbeitung oder ein Motiv, das das Modell hergibt. Wer diese Grenze akzeptiert, spart sich die Hälfte seiner Rechenzeit.

Und eines bleibt immer: der Zufall. Derselbe Prompt mit einem anderen Seed kann das Detail plötzlich richtig treffen, ohne dass du ein Wort geändert hast. Manchmal ist die schnellste Lösung kein besserer Prompt, sondern ein zweiter Versuch. Wer das Gesamtbild dahinter sucht – vom Diffusionsprinzip bis zur Governance –, findet es auf der Übersichtsseite [KI-Bildgenerierung](#).

## Häufige Fragen

### Warum ignoriert die KI Teile meines Prompts?

Meist aus einem von fünf Gründen: Dein Prompt enthält einen Widerspruch, ein Merkmal wird dem falschen Objekt zugeordnet, zwei ähnliche Dinge verschmelzen, du hast eine Verneinung

benutzt, oder der Trainingsbias des Modells ist stärker als deine Beschreibung. Prüfe in dieser Reihenfolge. Bleibt danach etwas übrig, ist es wahrscheinlich eine echte Modellgrenze.

### **Hilft es, den Prompt länger und genauer zu machen?**

Genauer ja, länger nein. Konkrete Angaben schlagen schmückende Wörter, aber mit jedem zusätzlichen Baustein verteilt sich die Aufmerksamkeit auf mehr Begriffe. Lange Prompts verschlimmern zudem das Verschmelzen von Konzepten, und was hinten steht, geht zuerst unter. Wenige, klare, widerspruchsfreie Bausteine schlagen die lange Wunschliste.

### **Warum landet ausgerechnet das im Bild, was ich ausschließen wollte?**

Weil das Wort in deinem Prompt steht und der Text-Encoder Verneinungen kaum auswertet. „Keine Kaffeetasse“ nennt die Kaffeetasse beim Namen. Formuliere positiv, was stattdessen zu sehen sein soll, oder nutze den Negativ-Prompt – sofern dein Modell einen mit echter Wirkung hat.

### **Warum vertauscht die KI Farben und Eigenschaften?**

Das ist Prompt-Bleeding: Merkmale färben im Text-Encoder auf benachbarte Begriffe ab, und die Zuordnung zum richtigen Objekt ist nur eine Wahrscheinlichkeit. Schreibe jedes Merkmal direkt neben sein Objekt, beschreibe Objekte in getrennten Blöcken und nutze bei CLIP-Modellen BREAK, um Blöcke sauber zu trennen.

### **Liegt es an mir oder am Modell?**

Faustregel: Widersprüche, Reihenfolge, Verneinungen und Zuordnung sind dein Part. Fehlendes Wissen, genaues Zählen, exakte Positionen und kaputte Hände sind Modellgrenzen. Der schnellste Weg zur Antwort sind die vier Tests: fünf Seeds, Prompt halbieren, Minimalprompt, anderes Modell. Wenn alle vier scheitern, liegt es am Modell.

### **Bekomme ich mit demselben Prompt immer dasselbe Problem?**

Nicht unbedingt. Der Seed bestimmt das Startrauschen, und ein anderer Seed kann ein störendes Detail zufällig richtig treffen. Bevor du den Prompt zerlegst, lohnt sich fast immer ein zweiter Durchlauf mit neuem Seed – das ist der billigste Test, den es gibt.

### **Helfen Ausrufezeichen, Großbuchstaben oder das Wort „wichtig“?**

Nein. Für den Text-Encoder sind das gewöhnliche Token ohne Sonderbedeutung. Sie verbrauchen Aufmerksamkeit und liefern nichts zurück. Mehr Gewicht bekommt ein Baustein über seine Position im Prompt und über die Gewichtungssyntax deiner Oberfläche, nicht über Nachdruck.

### **Warum funktioniert derselbe Prompt in einem anderen Werkzeug besser?**

Weil Oberflächen unterschiedlich mit deinem Text umgehen. Die Gewichtung in Klammern wird in der Automatic1111-Familie nach dem Multiplizieren normalisiert, in ComfyUI standardmäßig nicht – dieselbe 1.3 wirkt dort härter. Dazu kommen unterschiedliche Voreinstellungen bei Guidance, Sampler und Schrittzahl. Vergleiche nur mit identischen Werten und identischem Seed.

### **Warum werden aus drei Äpfeln vier?**

Zählen gehört zu den bekanntesten Schwächen dieser Modelle und wird in Prüfständen wie GenEval als eigene Disziplin gemessen. Zahlwörter sind schwache Signale, weil sie keine sichtbare Eigenschaft beschreiben. Bei kleinen Mengen bis etwa drei klappt es oft, darüber wird es Glückssache. Wenn die Anzahl zwingend stimmen muss, gib die Komposition vor oder setze das Bild aus mehreren Durchläufen zusammen.

## Wann lohnt sich ein LoRA statt weiterer Prompt-Arbeit?

Sobald dasselbe Motiv wiederkehrt und der Prompt es nie zuverlässig trifft: dein Produkt, eine bestimmte Person, ein Werkstattstil, eine Markenwelt. Ein trainierter Zusatz löst das dauerhaft, was ein immer längerer Prompt nur annähert. Der Aufwand liegt bei Tagen statt Minuten, rechnet sich aber ab der zweiten oder dritten Kampagne.

## Kann ich das Problem mit einem Bearbeitungsmodell umgehen?

Oft ja, und es ist meist der schnellere Weg. Statt ein ganzes Bild neu zu würfeln, gibst du einem Bearbeitungsmodell das fertige Bild plus eine Anweisung für genau die strittige Stelle. Der Rest bleibt erhalten. Für kleine Korrekturen ist Inpainting die ältere, aber immer noch zuverlässigste Variante.

## Muss ich kennzeichnen, dass ein Bild KI-erzeugt ist?

Für viele Einsatzzwecke ja. Seit dem 2. August 2026 gelten die Transparenzpflichten des EU AI Act, und unabhängig davon ist Kennzeichnung inzwischen Teil einer glaubwürdigen Kommunikation. Was konkret gilt und für wen, steht in [EU AI Act: Was seit dem 2. August 2026 gilt](#).

### Passend zum Thema

- › Wie ein Prompt aufgebaut wird und wo der Negativ-Prompt wirkt: [Prompt und Negativ-Prompt in der KI-Bildgenerierung](#)
- › Warum ein anderer Seed manchmal mehr hilft als ein besserer Prompt: [Der Seed bei der KI-Bildgenerierung](#)
- › Wie stark das Modell dem Text überhaupt folgt: [CFG-Scale: Guidance in KI-Bildgeneratoren](#)
- › Die hartnäckigste Detailschwäche im Einzelnen: [Warum KI keine Hände kann](#)
- › Welcher Modelltyp vor dir liegt: [Checkpoints, Modelle und VAE](#) · [Die 15 wichtigsten Modelle zur lokalen Bildgenerierung](#)
- › Wenn Prompt-Arbeit nicht mehr reicht: [Was ist ein LoRA?](#)
- › Grundlagen zum Verfahren dahinter: [Basiswissen KI-Bildgenerierung: Wie Diffusion funktioniert](#)
- › Wie viele Varianten pro Durchlauf sinnvoll sind: [batch size und Reproduzierbarkeit](#)
- › Bildformat und Bildausschnitt sauber festlegen: [Auflösung & Seitenverhältnis bei KI-Bildern](#)
- › Prompt-Arbeit direkt in Office: [Black Forest FLUX.2 in PowerPoint und Copilot](#)
- › Sprachmodelle im Unternehmen jenseits der Bildgenerierung: [Claude im Unternehmen](#)