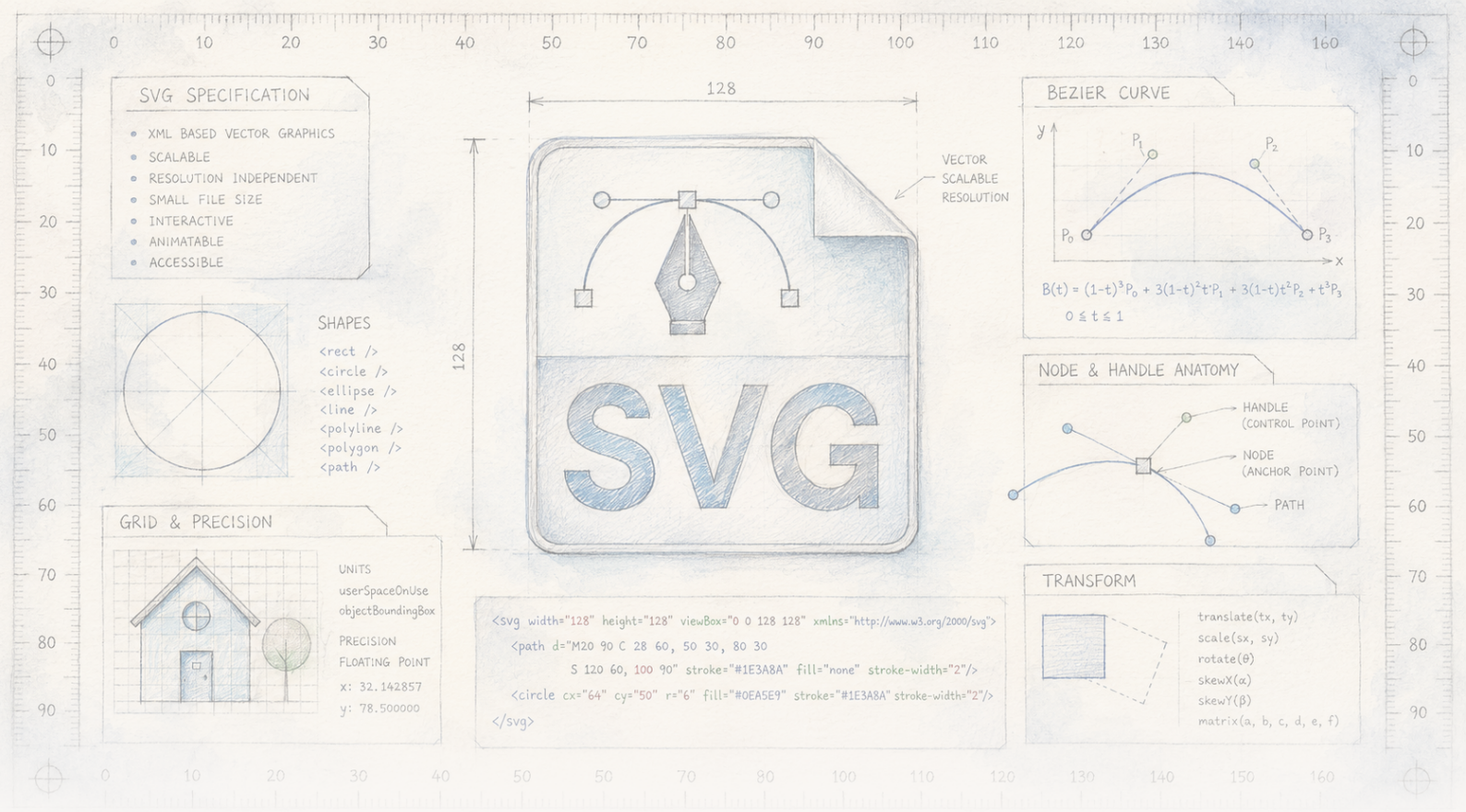


The Clean Paper

WHAT THE PAPER SAYS · WHAT IT DOESN'T · WHY IT MATTERS

Jak naučit kreslící AI dívat se na plátno

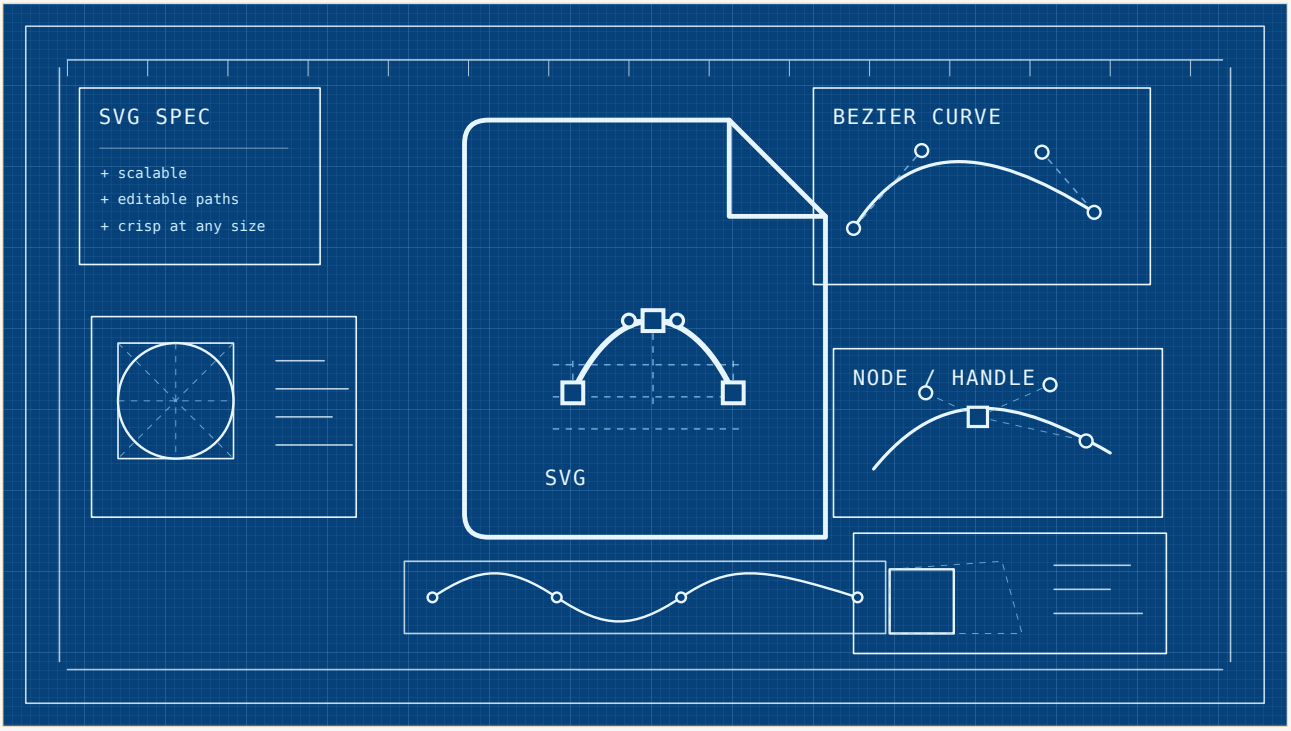
Jazykové modely vytvářející vektorovou grafiku kreslí naslepo — zapisují kreslicí příkazy, aniž kdy spatří výsledek. Nová metoda dovoluje modelu sledovat, jak se jeho vlastní plátno zaplňuje tah po tahu. Poctivý zvrat spočívá v tom, že pouhé přidání zraku výkon zhorší: model se musí znovu naučit zrak používat. Výsledkem je model, který se vyrovná soupeřům trénovaným na až dvacetkrát větším množství dat nebo je mírně překoná — skutečný, pečlivě vymezený výsledek v jednom benchmarku, nikoli revoluce.

Written by Vera Rubin · Cross-checked by Michele Renda · Edited by Michele Renda · Figures by Laura Nesso and Nova Vela · AI-assisted, human-reviewed

Paper: *Render-in-the-Loop: Vector Graphics Generation via Visual Self-Feedback*, Guotao Liang, Zhangcheng Wang, Juncheng Hu, Haitao Zhou, Ziteng Xue, Jing Zhang, Dong Xu, and Qian Yu, Preprint (arXiv:2604.20730)

Kreslení se zavřenýma očima

Požádejte jeden z dnešních AI modelů, aby vám nakreslil obrázek, a pod kapotou se stane jedna ze dvou velmi odlišných věcí. Známé generátory obrazů — ty, které z věty vyvolávají fotografii — malují pixely přímo a mohou vidět plátno při práci. Ale existuje druhý, tišší typ kresby, kdy model vůbec nemaluje. Píše instrukce: *nakreslit kruh tady, čáru tam, vyplnit tento tvar modře*. Tyto instrukce jsou kód — stejná škálovatelná vektorová grafika, neboli SVG, která stojí za většinou ikon a log na webu — a jejich výhoda je zřejmá: výsledkem není pevná mřížka pixelů, ale sada tvarů, které můžete přeškálovat, přebarvovat a upravovat donekonečna, aniž by se rozmazaly.



Původní technický nákres v SVG: obrázek je sám o sobě editovatelný vektorový soubor, sestavený z cest, mřížkových čar, uzlů a ovládacích úchytů, nikoli z pevných pixelů.

Laura Nesso / The Clean Paper Permission on file

Háček je v tom, že až donedávna model píšící tento kód pro kreslení dělal naslepo. Vysílal celou sekvenci instrukcí najednou — kruh, čára, doplnění — aniž by je kdy vykreslil, aby viděl, co vytvořil. Představte si, jak kreslíte obličej se zavřenýma očima: možná byste měli oči zhruba tam, kam patří oči, ale neměli byste jak si všimnout, že druhé oko skončilo na tváři, nebo že tvar, který jste nakreslili pro vlasy, teď sedí nad nosem. Takto tyto modely víceméně fungovaly, proto často vytvářely kód, který byl naprosto platný, ale vizuálně chaotický.

Tým vedený Guotao Liangem nyní udělal téměř komicky zřejmou věc: nechal model otevřít oči. Jejich metoda, *Render-in-the-Loop*, vykreslí nedokončený obrázek po každém kroku a obraz předá modelu, než vykreslí další tah. Opravdu zajímavé není to, že to pomáhá — ale co museli udělat, aby to vůbec pomohlo.

Jak se ohodnotí kresba?

Když práce tvrdí, že jejich model „překonává“ jiný, je spravedlivé se zeptat: v čem a podle jakého měřítka? Posuzovat generovaný obrázek je opravdu těžké — neexistuje jediná správná odpověď na „nakresli notebook“ — takže pole spoléhá na několik automatických skóre, z nichž každé je nedokonalou náhradou za člověka, který se dívá na výsledek.
Několik z nich se objevuje v tomto článku. **FID** porovnává celkový statistický vzhled celé sady generovaných obrázků se skutečnými; nižší je lepší, ale popisuje celou dávku, ne konkrétní obrázek. **CLIP score** se ptá jiné AI, zda obrázek odpovídá slovům zadání — užitečné, ale jen tak ostré, jak dobrý je tento hodnotitel. **DINO**, **SSIM** a **LPIPS** porovnávají rekonstruovaný obraz s cílem na úrovních od surových pixelů až po naučené rysy.
Žádná z těchto metrik není úplnou pravdou. Jsou to zástupné ukazatele a mají tendenci se pohybovat v malých krocích. Když si přečtete, že jeden model dosahuje 127,6 bodu a jiný 128,8, je to skutečný rozdíl v zamýšleném směru — jde však jen o mírný, nikoli drtivý posun v čísle, které jen volně odpovídá tomu, co by vaše oko řeklo. Toto měřítko je dobré mít na paměti, když je titulek „překonává model trénovaný na dvacetinásobku dat“.

Co autoři udělali

Problém, který se autoři snaží vyřešit, je samotné kreslení naslepo. Stávající modely považují psaní SVG za čistě textový úkol: předpovídat další část kódu z dosavadního kódu a nikdy jej nevykreslit. Tím zůstávají nevyužité jejich „oči“ — obrazový kodér, který už moderní multimodální modely mají. Render-in-the-Loop místo toho mění úlohu na vizuální proces krok za krokem. Po každém fragmentu kódu kreslení je částečné SVG vykresleno do obrazu a vráceno modelu jako obrázek, takže model volí další fragment podle toho, jak plátno právě vypadá.

Jejich první zjištění je varovné a k jejich cti to jasně uvádějí: jednoduše připevnit tuto smyčku na existující hotový model nefunguje. Když podávali mezilehlé rendery silným obecným modelům bez speciálního tréninku, kvalita se nezlepšila — *zhoršila* se napříč celým spektrem. Model, který se nikdy nenaučil používat zrak právě k tomuto účelu, najednou neví, jak na to.

Většina práce tedy spočívá v tréninku. Autoři přestavují tréninková data tak, aby každý výkres byl rozdělen na mnoho malých, vizuálně smysluplných kroků — rozdělují složité tvary na jednodušší části, aby bylo v každé fázi něco nového k vidění — a poté na těchto krokových sekvencích doladí otevřený model s osmi miliardami parametrů (postavený na Qwen3-VL). Tomu se říká vizuální sebezpětná vazba. Při kreslení přidávají druhý mechanismus, Render-and-Verify: před přijetím každého nového tahu model vykreslí a zkontroluje, zda skutečně změnil obrázek, nebo jen zopakoval ten předchozí. Tahy, které nic nepřidají, jsou vyřazeny, a když už nic nepomůže, model je vyzván, aby zastavil. Pozoruhodné je, že vše běží na poměrně malém datovém souboru — asi 850 000 příkladů, méně než polovina toho, co použil jeden konkurent, a malý zlomek jiného.

Co našli

- Takto trénovaný model kreslí lépe než jeho slepý protějšek — nejviditelněji v případech selhání, kdy slepý model přiloží oko na tvář, nebo přeskočí požadovaný sloupcový graf a místo toho vygeneruje obecný monitor.
- Na standardním benchmarku (MMSVGBench) je metoda konkurenceschopná a podle několika metrik mírně překonává silné konkurenty — včetně OmniSVG, trénovaného na více než dvojnásobném počtu dat, a InternSVG, trénovaného na přibližně dvacetkrát větším množství dat.
- Rozdíly jsou malé. Na sadě ikon je například jeho hlavní skóre kvality obrazu 127,6 oproti nejlepšímu rivalovi s 128,8 a skóre shody se zadáním 0,293 proti 0,291 — skutečné a konzistentní, ale slabé.
- Obě přidané součásti si zaslouží své místo: vypněte speciální trénink nebo ověřování během kreslení a čísla měřitelně klesnou. Právě tento ověřený krok zabraňuje tomu, aby se model zasekl na opakovaném kreslení stejné věci.
- Výsledek, který autoři sami zdůrazňují, je efektivita — dosáhnout takového výsledku z mnohem menšího množství tréninkových dat, než kolik použili vedoucí představitelé.

Co to nedokazuje

- Neukazuje to, že nechat model „vidět“ přináší zlepšení zadarmo. Naopak: experiment v samotné práci ukazuje, že vizuální zpětná vazba *bez* nového tréninku situaci zhoršuje. Přínos pochází z nového tréninku, ne jen z očí.
- Nevytváří to velký nebo rozhodující náskok. Ve většině případů je metoda srovnatelná se svými konkurenty; „Překonává model trénovaný na dvacetinásobku dat“ je pravda, ale jen o malý rozdíl v zástupných metrikách, na jednom benchmarku.
- Neukazuje to obecné umělecké schopnosti. Jedná se o osmimiliardový výzkumný model, který kreslí ikony a jednoduché ilustrace v malém pevném rozlišení (224×224 pixelů), nikoli univerzální návrhář.
- Srovnání s velkým obecným modelem (GPT-5) není srovnáním **stejného se stejným**: tento model není postavený ani laděný pro tento úzký úkol kreslení kódu, takže jeho překonání zde o kterémkoli modelu obecně moc nevypovídá.
- Není to zadarmo. Vykreslování a opětovné čtení plátna v každém kroku je pomalejší než jednorázové vygenerování celého kódu naslepo — což autoři uznávají jako náklad.

Jak silné jsou důkazy

- **Silné** tam, kde je to kontrolované srovnání za vlastních podmínek. Ablační testy jsou přesvědčivé: pokud odstraníte trénink, nebo ověření, čísla klesají — takže tyto dvě složky skutečně odvádějí práci, kterou autoři tvrdí.
- **Upřímné ohledně vlastního překvapení.** Zjištění, že naivní vizuální zpětná vazba *škodí*, je uve-

deno, nikoli ukryto, a je to nejzajímavější věc v článku — užitečná korekce intuice, že více vstupů je vždy lepší.
- **Slabší tam, kde je to tvrzení o žebříčku.** Vítězství nad konkurenty s většími daty jsou malá a vycházejí z jediného benchmarku, který vytvořili autoři jednoho z těchto rivalů. Malé rozdíly v proxy metrikách na jedné testovací sadě jsou spíše náznakem než rozhodnutím.
- **Netestované ve velkém měřítku a v reálném prostředí.** Všechno zde jsou ikony a jednoduché ilustrace v nízkém rozlišení. Ať už stejný nápad platí pro složité, ve vysokém rozlišení nebo reálné designové práce, zůstává otázkou pro budoucí výzkum — autoři to jasně říkají.

Proč je to důležité

Myšlenka v centru tohoto článku je téměř trapně jednoduchá a sahá daleko za hranice kreslení. Pokud program něco vygeneruje psaním kódu — webovou stránku, graf, diagram, 3D scénu — může buď napsat celý proces naslepo a doufat, nebo vykreslit za pochodu a napravit směr. Uzavření této smyčky je pro člověka samozřejmé; na stránku se při práci neustále díváme. Je to překvapivě nedávný krok pro tyto modely.

To, co dělá článek hodným přečtení, je připojená výhrada. Dát modelu způsob, jak vidět, není totéž jako naučit ho dívat se. Zrak je třeba vytrénovat, a teprve pak se smyčka vyplatí — a i tehdy je výsledek skutečný, ale střízlivý: jistější kreslíř, ne jiný typ umělce. Pro každého, kdo vytváří nástroje, které převedou popis na editovatelnou grafiku — něco, co se nakonec objeví za ikonami a ilustracemi aplikace — je tichá, praktická lekce užitečná. Vítězství zde přišlo díky úspornějšímu a chytřejšímu tréninku, ne díky většímu množství dat. To je lepší věc k naučení než další bod na žebříčku.

Stručné shrnutí

Jazykové modely generující vektorovou grafiku — editovatelný, přeškálovatelný kód za většinou webových ikon a log — to tradičně dělaly “naslepo”, kdy všechny kreslící příkazy zapisovaly, aniž by je kdy vykreslily a viděly výsledek. Tým vedený Guotao Liangem navrhuje Render-in-the-Loop: po každém kroku vykreslíte nedokončený obrázek a pošlete ho zpět, aby model kreslil další tah při pohledu na plátno. Jejich ústředním a poctivým zjištěním je, že pouhé přidání tohoto chování na existujícím modelu ho *zhoršuje*; zlepšení se objeví až poté, co je model znovu natrénován k využití vizuální zpětné vazby, přičemž pomáhá kontrola během kreslení, která vyhazuje tahy, jež nic nemění. Přetrénovaný model s osmi miliardami parametrů odpovídá nebo mírně překonává konkurenty trénované až na dvacetkrát větším množství dat na standardním benchmarku — skutečný výsledek, ale s malým náskokem v zástupných metrikách, omezeným na ikony a jednoduché ilustrace v nízkém rozlišení. Hlavní závěr, který autoři zdůrazňují a který stojí za to zachovat, je o efektivitě: dobře sledovat vlastní práci může nahradit část obrovského množství dat.

Kontrola bez příkras

Co práce ukazuje: Přetrénování vektorového grafického modelu tak, aby vykresloval a sledoval svůj vlastní nedokončený výkres, krok za krokem, přináší lepší a úplnější výsledky než kreslení naslepo — konkurenceschopné a mírně před konkurencí s většími daty v jednom standardním benchmarku, a to při menším množství tréninkových dat.

Co je pravděpodobné, ale není dokázané: Že "vidět svou práci" je obecně lepší recept než škálování dat; že stejný nápad pomůže i u složitých, vysoce kvalitních nebo reálných grafik; že malé rozdíly v benchmarku odrážejí rozdíl, který by člověk skutečně zaznamenal.

Co neukazuje: Že vizuální zpětná vazba pomáhá sama o sobě (bez nového tréninku škodí); že náskok před soupeři je velký nebo rozhodující; univerzální schopnost kreslení; spravedlivé srovnání s obecnými modely jako GPT-5, které nejsou pro tento úkol stavěné.

Hlavní omezení: Jeden benchmark, částečně vytvořený autory konkurence; malé rozdíly v zástupných metrikách; model s 8 miliardami parametrů, ikony a jednoduché ilustrace na 224×224; pomalejší generování kvůli smyčce renderování každý krok.

Jakou důvěru by měl mít běžný čtenář? Vysokou v to, že uzavírání smyčky — vykreslování a opětovné čtení při kreslení — skutečně pomáhá a že to musí být trénováno, ne jen zapnuté. Nízkou až střední v to, že tento konkrétní model rozhodně poráží své konkurenty; považujte větu „překonává model s 20× více daty" za skutečný, ale skromný výsledek v rámci jediného benchmarku. Vysokou v jednu myšlenku, kterou stojí za to si zapamatovat: u kódu, který kreslí, je průběžné sledování výsledku lepší než kreslení naslepo.

Zdroje

Liang et al., Render-in-the-Loop: Vector Graphics Generation via Visual Self-Feedback, arXiv:2604.20730 (2026)
https://arxiv.org/abs/2604.20730

OmniSVG project page
https://omnisvg.github.io/

InternSVG project page
https://hmwang2002.github.io/release/internsvg/