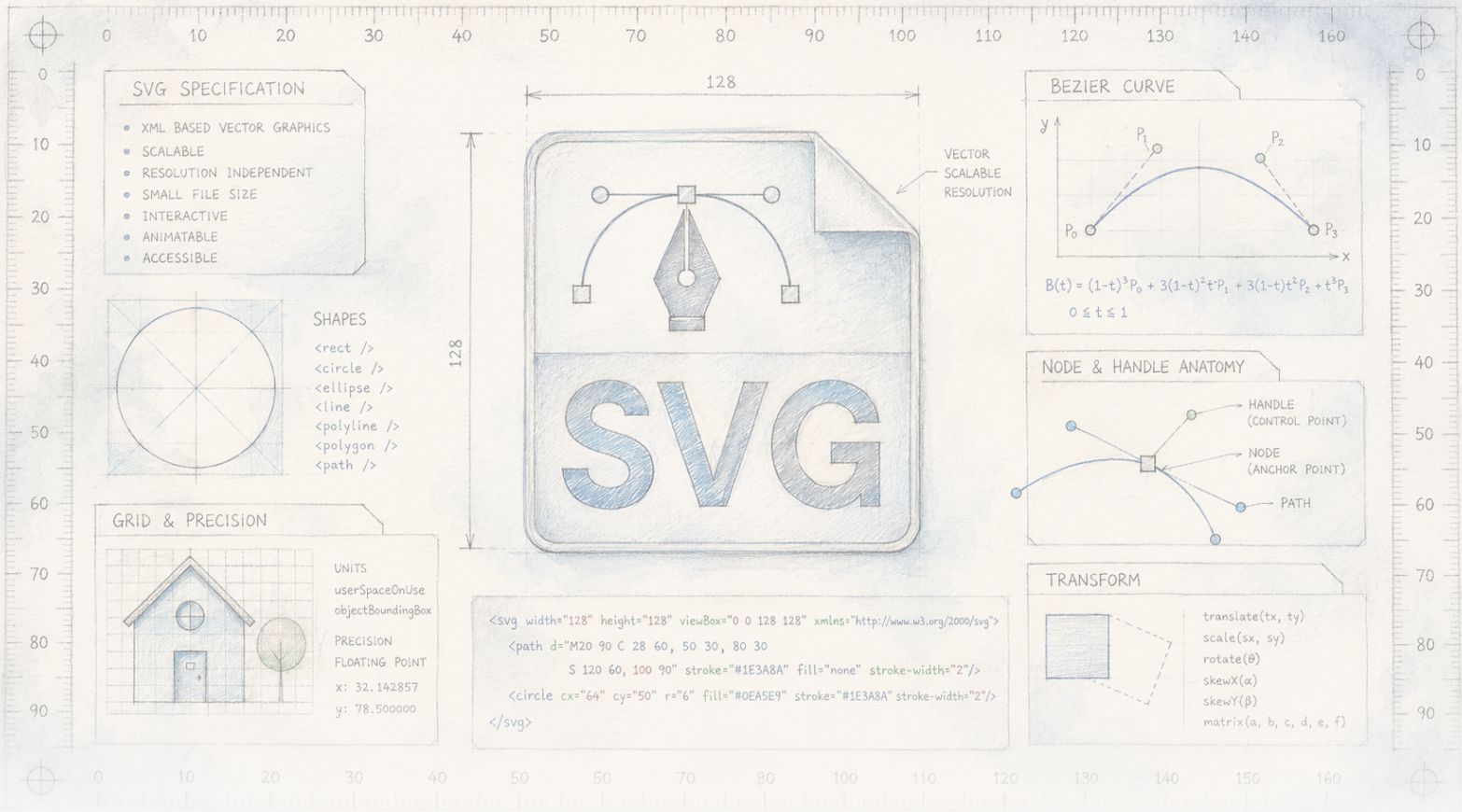


The Clean Paper

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

Навучыць AI-мастака глядзець на старонку

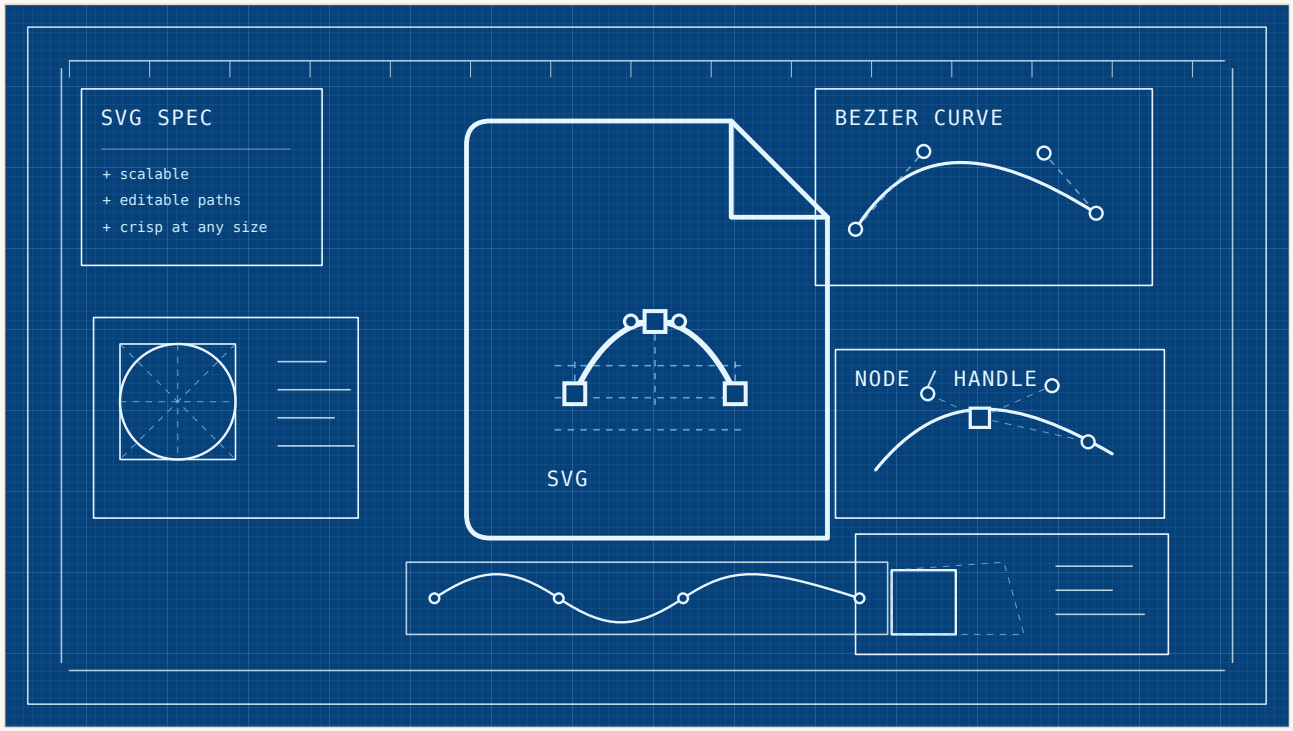
Моўныя мадэлі, якія генеруюць вектарную графіку, звычайна робяць гэта ўсляпую — пішуць каманды малявання, ні разу не ўбачыўшы вынік. Новы метад дазваляе мадэлі назіраць, як палатно запаўняецца штрых за штрыхом, і сумленны паварот у тым, што проста даць ёй вочы робіць вынік горшым: мадэль трэба перанавучыць імі карыстацца. У выніку яна параўноўваецца або крыху пераўзыходзіць канкурэнтаў, навучаных на даных аб'ёмам да дваццаці разоў большым, — рэальны, але асцярожны вынік аднаго бенчмарку, а не рэвалюцыя.

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)

Маляваць з заплюшчанымі вачыма

Папрасіце адну з сучасных AI-мадэляў намаляваць карцінку — і пад капотам адбудзецца адна з дзвюх вельмі розных рэчаў. Знаёмыя генератары выяў, якія ствараюць фатаграфію паводле сказа, непасрэдна «малююць» пікселі і бачаць палатно ў працэсе працы. Але ёсць і другі, цішэйшы від малявання, дзе мадэль увогуле нічога не малюе. Яна піша інструкцыі: *намалюй тут круг, там лінію, залі гэтую фігуру сінім*. Гэтыя інструкцыі — код, той самы Scalable Vector Graphics, або SVG, які ляжыць у аснове большасці іконак і лагатыпаў у вэбе. Яго перавага рэальная: вынік — не фіксаваная сетка пікселяў, а набор фігур, якія можна бясконца маштабаваць, перафарбоўваць і рэдагаваць без размыцця.



Арыгінальная SVG-ілюстрацыя ў стылі тэхнічнага чарцяжа: сама выява ўяўляе сабой рэдагуемы вектарны файл, пабудаваны з контураў, ліній сеткі, вузлоў і кіруючых ручак, а не з фіксаваных пікселяў.

Laura Nesso / The Clean Paper Permission on file

Праблема ў тым, што да нядаўняга часу мадэль, якая пісала такі код для малюнка, рабіла гэта ўсляпую. Яна выдавала ўсю паслядоўнасць каманд за адзін праход — круг, лінія, заліўка — і ні разу не рэндэрыла вынік, каб паглядзець, што атрымалася. Уявіце, што малюеце твар з заплюшчанымі вачыма: вы, магчыма, прыкладна паставіце вочы туды, дзе ім належыць быць, але не заўважыце, што другое апынулася на шчацэ або што фігура валасоў цяпер перакрывае нос. Прыкладна так гэтыя мадэлі і працавалі, таму нярэдка ствaралі цалкам карэктны код, які візуальна ператвараўся ў беспарадак.

Каманда пад кіраўніцтвам Guotao Liang зрабіла амаль камічна відавочную рэч: дазволіла мадэлі расплюшчыць вочы. Іх метад *Render-in-the-Loop* пасля кожнага кроку рэндэрыць незавершаны малюнак і вяртае гэтую карцінку мадэлі перад наступным

штрыхом. Па-сапраўднаму цікава не тое, што гэта дапамагае, а тое, што ім давялося зрабіць, каб гэта ўвогуле пачало дапамагаць.

Як ацэньваюць малюнак?

Калі артыкул кажа, што яго мадэль «перамагае» іншую, справядліва спытаць: перамагае ў чым і паводле якога вымярэння? Ацэньваць згенераваную выяву сапраўды цяжка — адзінага правільнага адказу на просьбу «намалюй ноўтбук» няма, — таму галіна абапіраецца на некалькі аўтаматычных метрык, кожная з якіх толькі недасканала замяняе чалавека, які глядзіць на вынік.
У гэтай працы выкарыстоўваюцца некалькі такіх метрык. **FID** параўноўвае агульныя статыстычныя ўласцівасці цэлай партыі згенераваных выяў з рэальнымі; менш — лепш, але гэта ацэнка партыі, а не асобнай карцінкі. **CLIP score** пытаецца ў іншай AI-мадэлі, наколькі выява адпавядае тэкставаму запыту — карысна, але толькі настолькі, наколькі добры гэты суддзя. **DINO**, **SSIM** і **LPIPS** параўноўваюць рэканструяваную выяву з мэтавай на ўзроўнях ад сырых пікселяў да навучаных прыкмет.
Ніводная з гэтых метрык не з'яўляецца ісцінай. Гэта проксі-паказчыкі, і яны часта змяняюцца малымі крокамі. Калі вы чытаеце, што адна мадэль атрымала 127,6, а другая 128,8, гэта рэальная розніца ў патрэбны бок — але невялікі зрух, а не абвал, прычым у ліку, які толькі прыблізна адлюстроўвае тое, што сказаў бы ваш вачэй. Гэта варта памятаць, калі загаловак гучыць як «перамагае мадэль, навучаную на ў дваццаць разоў большым аб'ёме даных».

Што зрабілі аўтары

Аўтары хацелі выправіць менавіта гэтае сляпое маляванне. Папярэднія мадэлі разглядалі напісанне SVG як чыста тэкставую задачу: прадказаць наступны кавалак кода па ўжо напісаным і ніколі яго не рэндэрыць. Праз гэта без справы заставаліся магутныя «вочы» — візуальны энкодэр, які ўжо ёсць у сучасных мультымадальных мадэляў. Render-in-the-Loop перабудоўвае задачу ў пакрокавы візуальны працэс. Пасля кожнага фрагмента кода частковы SVG рэндэрыцца як выява і падаецца назад мадэлі, так што наступны фрагмент яна выбірае, сапраўды гледзячы на бягучае палатно.

Іх першы вынік — перасцярога, і, да іх гонару, яны паведамляюць яго наўпрост: проста прымацаваць гэты цыкл да гатовай мадэлі не працуе. Калі даследчыкі падавалі прамежкавыя рэндэры моцным універсальным мадэлям без спецыяльнага навучання, якасць не палепшылася — яна *пагоршылася* па ўсіх паказчыках. Мадэль, якую ніколі не вучылі выкарыстоўваць вочы менавіта для гэтай задачы, не пачынае раптам умець гэта рабіць.

Таму большая частка працы звязаная з навучаннем. Аўтары перабудоўваюць навучальныя даныя так, каб кожны малюнак быў разбіты на мноства невялікіх, візуальна асэнсаваных

крокаў — складаныя фігуры дзеляць на прасцейшыя часткі, каб на кожнай стадыі з'яўлялася нешта новае для прагляду, — а затым данавучаюць адкрытую мадэль з васьмю мільярдамі параметраў (на аснове Qwen3-VL) на гэтых пакрокавых паслядоўнасцях. Гэта яны называюць Visual Self-Feedback. Дадаецца і другі механізм падчас малявання — Render-and-Verify: перш чым прыняць кожны новы штрых, мадэль рэндэрыць яго і правярае, ці сапраўды ён змяніў карцінку або толькі паўтарыў папярэдняе. Штрыхі, якія нічога не дадаюць, адкідаюцца, а калі далейшыя крокі ўжо не дапамагаюць, мадэлі прапануюць спыніцца. Характэрна, што ўсё гэта працуе на даволі невялікім наборы — прыкладна 850 000 прыкладаў, менш за палову таго, што выкарыстоўваў адзін канкурэнт, і малая доля даных іншага.

Што яны выявілі

- Навучаная такім чынам мадэль малюе лепш за сваю «сляпую» версію — асабліва прыкметна на тыповых памылках, калі сляпая мадэль змяшчае вока на шчацэ або ігнаруе запытаную слупковую дыяграму і выдае замест яе звычайны манітор.
- На стандартным бенчмарку MMSVGBench метад канкурэнтаздольны і па некалькіх метрыках крыху апярэджвае моцных супернікаў — у прыватнасці OmniSVG, навучаны на больш чым удвая большым наборы даных, і InternSVG, навучаны прыкладна на ў дваццаць разоў большым.
- Адрыў невялікі. На наборы іконак, напрыклад, галоўная метрыка якасці выявы складае 127,6 супраць 128,8 у найлепшага канкурэнта, а паказчык адпаведнасці запыту — 0,293 супраць 0,291: рэальныя і паслядоўныя розніцы, але малыя.
- Абодва дададзеныя кампаненты апраўдваюць сваё месца: адключыце спецыяльнае навучанне або праверку падчас малявання — і паказчыкі прыкметна падаюць. Праверка асабліва добра не дае мадэлі захраснуць, зноў і зноў перамалёўваючы адно і тое ж.
- Вынік, які найбольш падкрэсліваюць самі аўтары, — эфектыўнасць: такога ўзроўню ўдалося дасягнуць з значна меншай колькасцю навучальных даных, чым выкарыстоўвалі лідары.

Чаго гэта не даказвае

- Гэта **не паказвае**, што даць мадэлі «зрок» — бясплатная перамога. Наадварот: уласны эксперымент артыкула паказвае, што візуальная зваротная сувязь *без* перанавучання пагаршае вынік. Выйгрыш прыходзіць ад навучання, а не ад саміх вачэй.
- Гэта **не ўсталёўвае вялікай або вырашальнай перавагі**. Па большасці метрык метад ідзе ўпоравень з канкурэнтамі; «перамагае мадэль з 20× большым аб'ёмам даных» — праўда, але гаворка ідзе пра малыя зрухі проксі-метрык на адным бенчмарку.

- Гэта **не дэманструе агульнай мастацкай здольнасці**. Гэта даследчая мадэль на восем мільярдаў параметраў, якая малюе іконкі і простыя ілюстрацыі пры невялікай фіксаванай раздзяляльнасці 224×224 пікселяў, а не ўніверсальны дызайнер.
- Параўнанне з вялікай універсальнай мадэллю GPT-5 **не з'яўляецца прамым параўнаннем аднолькавых рэчаў**: GPT-5 не створаная і не наладжаная для гэтай вузкай задачы малявання кодам, таму перамога тут мала кажа пра любую з мадэляў у цэлым.
- Гэта **не бясплатна з пункту гледжання вылічэнняў**. Рэндэрыць і нанова чытаць палатно на кожным кроку павольней, чым адзін раз усляпую выдаць увесь код, — і аўтары гэта прызнаюць.

Наколькі моцныя доказы

- **Моцныя** там, дзе ёсць кантраляванае параўнанне на ўласных умовах. Абляцыі чыстыя: прыбярыце навучанне або праверку — паказчыкі падаюць, значыць, абедзве часткі сапраўды выконваюць заяўленую працу.
- **Сумленныя адносна ўласнага сюрпрызу.** Вынік, што наіўная візуальная зваротная сувязь *шкодзіць*, не схаваны, а паведамляецца наўпрост; гэта, бадай, самая цікавая частка артыкула і карысная папраўка да інтуіцыі, нібы больш уваходнай інфармацыі заўсёды лепш.
- **Тонкія як сцвярджэнне пра месца ў рэйтынгу.** Перамогі над канкурэнтамі з большымі наборамі даных невялікія і атрыманыя на адным бенчмарку, створаным аўтарамі адной з канкуруючых сістэм. Малыя адрывы ў проксі-метрыках на адным тэставым наборы — намёк, а не канчатковы прысуд.
- **Не правераныя ў вялікім маштабе і рэальнай працы.** Тут толькі іконкі і простыя ілюстрацыі нізкай раздзяляльнасці. Ці спрацуе тая ж ідэя для складанай, высокадэталёвай або рэальнай дызайнерскай задачы, застаецца будучай працай — самі аўтары гэта кажуць.

Чаму гэта важна

Ідэя ў цэнтры гэтага артыкула амаль няёмка простая і выходзіць далёка за межы малявання. Калі праграма генеруе нешта, запісваючы код — вэб-старонку, графік, дыяграму, 3D-сцэну, — яна можа альбо напісаць усё ўсляпую і спадзявацца, альбо рэндэрыць па ходзе працы і карэктаваць курс. Для чалавека замкнуць гэты цыкл натуральна: мы пастаянна пазіраем на старонку. Для такіх мадэляў гэта дзіўна нядаўні прыём.

Каштоўнасць артыкула — у зорачцы побач з гэтай ідэяй. Даць мадэлі магчымасць бачыць — не тое самае, што навучыць яе глядзець. «Вочы» трэба трэніраваць, і толькі тады цыкл акупляецца — ды і тады выйгрыш рэальны, але ўмераны: больш надзейны

чарцёжнік, а не іншы від мастака. Для тых, хто будуе інструменты, якія ператвараюць апісанне ў рэдагуемую графіку — менавіта такую, якая потым становіцца іконкамі і ілюстрацыямі праграм, — карысны менавіта гэты ціхі практычны ўрок. Выйгрыш тут прыйшоў ад таннейшага і разумнейшага навучання, а не ад большага аб'ёму даных. Гэта каштоўней за яшчэ адзін бал у табліцы лідараў.

Коратка

Моўныя мадэлі, якія генеруюць вектарную графіку — рэдагуемы, маштабуемы код, што ляжыць у аснове большасці вэб-іконак і лагатыпаў, — традыцыйна рабілі гэта «ўсляпую»: пісалі ўсе каманды малявання, ні разу не рэндэрыўшы вынік. Каманда пад кіраўніцтвам Guotao Liang прапануе Render-in-the-Loop: пасля кожнага кроку рэндэрыць незавершаны малюнак і падаваць яго назад, каб наступны штрых мадэль рабіла, гледзячы на палатно. Іх галоўны і сумленны вынік: простае даданне гэтага механізму да існуючай мадэлі робіць яе *горшай*; выйгрыш з'яўляецца толькі пасля перанавучання на выкарыстанне візуальнай зваротнай сувязі разам з праверкай падчас малявання, якая адкідае штрыхі, што нічога не змяняюць. Перанавучаная мадэль на восем мільярдаў параметраў на стандартным бенчмарку параўноўваецца або крыху пераўзыходзіць канкурэнтаў, навучаных на даных аб'ёмам да дваццаці разоў большым, — сапраўдны вынік, але з невялікімі адрывамі ў проксі-метрыках, на іконках і простых ілюстрацыях нізкай раздзяляльнасці. Выснова, якую падкрэсліваюць аўтары і якую варта захаваць, датычыцца эфектыўнасці: уменне добра бачыць уласную працу часткова замяняе простае нарошчванне маштабу даных.

Праверка без ажыятажу

Што паказвае артыкул: Перанавучанне мадэлі вектарнай графікі так, каб яна крок за крокам рэндэрыла і разглядала ўласны незавершаны малюнак, дае лепшыя і больш поўныя вынікі, чым сляпое маляванне, — канкурэнтныя і крыху лепшыя, чым у супернікаў з большымі наборамі даных на адным стандартным бенчмарку, пры меншай колькасці навучальных даных.

Што праўдападобна, але не даказана: Што «глядзець на ўласную працу» ў цэлым лепшая стратэгія, чым проста маштабаваць даныя; што тая ж ідэя дапаможа са складанай, высокадэталёвай або рэальнай графікай; што малыя розніцы ў бенчмарку адпавядаюць розніцы, якую чалавек сапраўды заўважыць.

Чаго праца не паказвае: Што візуальная зваротная сувязь дапамагае сама па сабе (без перанавучання яна шкодзіць); што перавага над канкурэнтамі вялікая або вырашальная; агульнай здольнасці маляваць; справядлівага прамога параўнання з універсальнымі мадэлямі накшталт GPT-5, якія не стваралісяся для гэтай задачы.

Асноўныя абмежаванні: Адзін бенчмарк, часткова створаны аўтарамі канкуруючай сістэмы; малыя адрывы ў проксі-метрыках; мадэль 8B, іконкі і простыя ілюстрацыі 224×224; больш павольная генерацыя праз цыкл рэндэрынгу на кожным кроку.

Колькі даверу варта мець звычайнаму чытачу? Высокі да таго, што замыканне цыклу — рэндэрыць і перачытваць вынік у працэсе — сапраўды дапамагае і гэтаму трэба навучаць, а не проста ўключыць. Нізкі або ўмераны да таго, што менавіта гэтая мадэль вырашальна перамагае канкурэнтаў; фразу «перамагае пры 20× меншым аб'ёме даных» варта ўспрымаць як рэальны, але сціплы вынік аднаго бенчмарку. Высокі да адной ідэі, якую варта запомніць: для кода, які малюе, глядзець у працэсе лепш, чым маляваць усляпую.

Крыніцы

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/