

2. B. Dmitry. Open beagle: a generic framework for evolutionary computations // Genetic Programming and Evolvable Machines. - 2011. - Vol.12. - P.329–331.
3. P. García-Sánchez, A. Mora, M.G. Arenas. Developing services in a service oriented architecture for evolutionary algorithms // Conference: Proceedings of the 15th annual conference companion on Genetic and evolutionary computation. - 2013. - P.1341-1348.

УДК 004.93

*ЛИННИК В. В.,
СИРОТА О.П.*

ПРАКТИЧНЕ ЗАСТОСУВАННЯ МЕТОДУ ГЕНЕРАЦІЇ ТЕКСТУР ОБ'ЄКТІВ В РОЗРІЗІ ЗАДАЧІ ТРИВИМІРНОЇ РЕКОНСТРУКЦІЇ

В даній статті було розглянуто практичне застосування методу реконструкції текстур об'єктів з визначенням положення камери, розміщенням об'єктів та показників освітлення зображення на прикладі задачі реконструкції об'єктів з єдиного зображення. Описано формальну процедуру проведення реконструкції текстур об'єктів та спосіб оцінки результатів, показано застосування м'якого растеризатора для контрольованого процесу рендерингу. На прикладі показано результати виконання даної реконструкції та проведено якісну оцінку результатів генерації.

КЛЮЧОВІ СЛОВА: 3D МОДЕЛІ, РЕКОНСТРУКЦІЯ ЗОБРАЖЕНЬ, КОМП'ЮТЕРНИЙ ЗІР, КОМП'ЮТЕРНА ГРАФІКА, НЕЙРОННІ МЕРЕЖІ

The subject of the article is an application of the method of texture object reconstruction with the determination of the camera position, placement of objects and indicators of image illumination on the example of the problem of single-view object reconstruction. It describes the formal procedure for texture object reconstruction and evaluating results, shows using soft rasterizer for the supervised process of rendering. The example shows the results of the reconstruction and a qualitative assessment of the generation results.

KEYWORDS: 3D MODELS, IMAGE RECONSTRUCTION, COMPUTER VISION, COMPUTER GRAPHICS, NEURAL NETWORKS

1. Вступ

Розуміння та реконструкція тривимірних сцен та структур з двовимірних зображень завжди було однією з найбільших цілей в задачах комп'ютерного зору. Ключем до відновлення тривимірного об'єкту є отримання характеристик моделі з кожного пікселю зображення. Останні роки, розробники досягли вражаючих результатів реконструкції моделей, однак їх роботи були сфокусовані на проблемі реконструкції геометрії та форм об'єктів, в той час як реконструкція текстур не отримала значної уваги спільноти, незважаючи на свою велику значимість для практичного застосування генерованих моделей. Тому, в даній статті розглянуто практичне застосування методу генерації текстур високої якості на прикладі

задачі генерації тривимірних моделей з єдиного зображення.

2. Проблема генерації текстур об'єктів

Попередні підходи до реконструкції текстури з єдиного зображення відрізняються від представлення інформації про текстуру на зображенні. Одним із існуючих підходів є використання об'ємних воксельних репрезентацій моделей для виконання кольорової тривимірної реконструкції. Однак, через обчислювальну складність представлення вокселів моделі, яка кубічно зростає з роздільною здатністю, їх моделі обмежені досить низькою роздільною здатністю (зазвичай 32^3 або 64^3) і, отже, не можуть відобразити якісні деталі. Альтернативне представлення текстури складається з двовимірного атласу текстури

та параметризованого представлення моделі з використанням UV-перетворення, яке встановлює відповідність між координатами на поверхні тривимірного об'єкту з зображенням текстури. Проте, сучасні підходи, засновані на представленнях сітки (повноцінних моделей), зазвичай передбачають відому топологію та попередньо визначену шаблонну сітку, яка обмежує ці підходи конкретними категоріями об'єктів, такими як птахи, обличчя, автомобілі. Прогнозування текстури для довільних фігур довільного класу без

шаблонної сітки залишається проблемою в галузі тривимірної реконструкції.

Зокрема, основним фактором, що обмежує якість результатів зазначених вище методів – це використання представлення текстур у вигляді, що не дозволяє виконувати якісну реконструкцію, методи що застосовуються лише до певного класу об'єктів, а також неможливість якісно порівняти згенерований результат з вхідним зображенням через відсутність даних про положення об'єкту на зображенні.

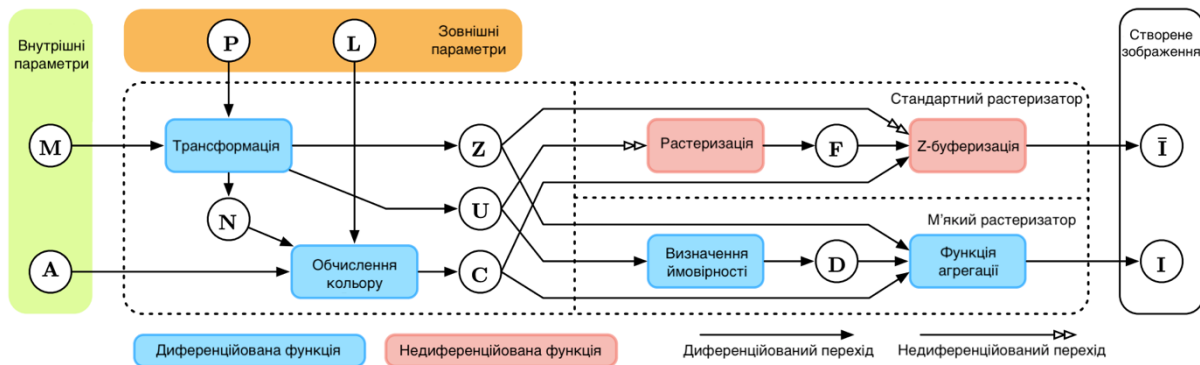


Рис. 1 Порівняння стандартного процесу растеризації та за допомогою м'якого растеризатору

3. Постановка задачі реконструкції текстур

Для реалізації задачі генерації текстур необхідно для заданого вхідного зображення об'єкту виконати його перетворення на модель у вигляді тривимірної сітки, визначити положення об'єкту на зображенні, його кут розміщення та коефіцієнт викривлення та після провести операції генерації та накладання текстури на об'єкт, при чому, текстурованими повинні бути не лише видимі частини моделі на зображенні, а модель повністю, та вираз $\|\hat{I} - I\|$ приймає мінімально можливе значення. При цьому, $\hat{I}$ – зображення, отримане шляхом рендерингу тривимірної моделі з накладенням текстур та параметрами положення об'єкту та камери, I – вхідне зображення об'єкту. Тобто, головна вимога до генератора текстур, це їх точне відтворення вхідного зображення.

4. Роль та місце рендерингу при генерації текстур

Використовуючи вхідні дані, нейромережа здатна згенерувати певну текстуру для об'єкту, проте для виконання валідації

згенерованих даних в задачі реконструкції моделей з єдиного зображення може виступати лише вхідне зображення, відповідно до цього, єдиним методом якісного знаходження точності створеної текстури є порівняння вхідного та отриманого процедурою рендерингу зображень.

Однак, стандартний процес візуалізації моделей є досить складним та недиференційованим, що підвищує складність розрахунків та зменшує гнучкість її використання для даної задачі. Зокрема, рендеринг складається з багатьох процесів растеризації, що перетворює дискретну вибірку точок моделі в частину вихідного зображення, проте даний процес є нелінійним та складним, а також може спричинити проблеми при заданні та підборі параметрів для максимальної відповідності вхідному зображенню, таким чином це може призвести до неконтрольованого процесу оптимізації текстур, що в свою чергу призведе до отримання помилкових текстур та невідповідність обрахованої похибки генерації реальним результатам.

Натомість, використання м'якого растеризатору[1] надає мережі гнучкі властивості з можливістю впливати на отриманий результат в реальному часі під час генерації. Як видно на рис. 1, розглянуто зовнішні параметри (камера P та показники кольору L), що визначають параметри середовища об'єкту, а також внутрішні параметри (структура моделі M та параметри вигляду A кожної вершини сітки, включаючи колір та матеріал), що описує параметри моделі. Слідуючи стандартному процесу

рендерингу система отримує нормаль моделі N , координати просторового зображення U та глибину зображення Z , що залежить від місця знаходження камери та отримана шляхом трансформації вхідних параметрів геометрії M , залежно від камери P . Колір визначається шляхом обчислення даних даними набором характеристик $\{A, N, L\}$. Обидва модулі (обчислення кольору та трансформації моделі) мають споріднену структуру та є повністю диференційованими. Однак, наступні операції,

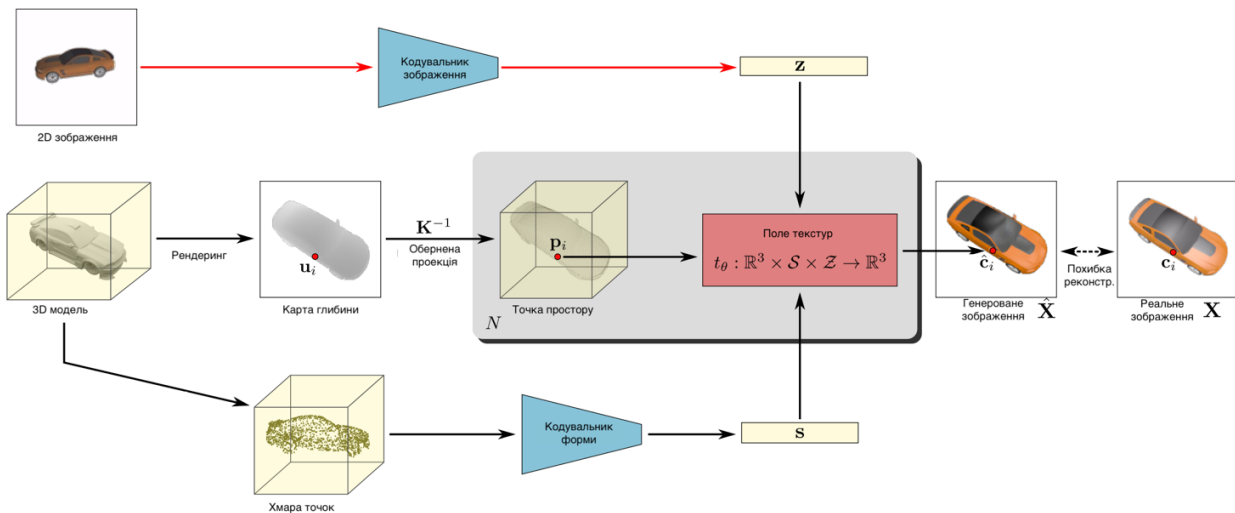


Рис. 3 Огляд формальної процедури виконання реконструкції текстур моделей

що включають растеризацію та z-буферизацію в стандартному процесі (червоні блоки на рис. 1) є недиференційовані через використання операцій дискретного розміщення U та Z .

На відміну від стандартного процесу, операція растеризації для м'якого растеризатору розглядається як бінарне маскування, що визначається як залежність між пікселями та поверхнями, коли z-буферизація поєднує результат растеризації F в один прохід базуючись на даних глибини, тоді як процедура м'якого рендеру базується на використанні карти ймовірностей D , що обраховує ймовірність кожного пікселя результуючого зображення належати певній площині моделі, після чого функція агрегації поєднує значення кольорів зі значеннями з карти ймовірностей, враховуючи глибину кожної площини моделі, таким чином процес рендерингу проходить без дискретних функцій та являє собою потік диференційованих функцій, робота яких контролюється коефіцієнтами впливу та не

потребує зміни алгоритму для утворення нового представлення моделі.

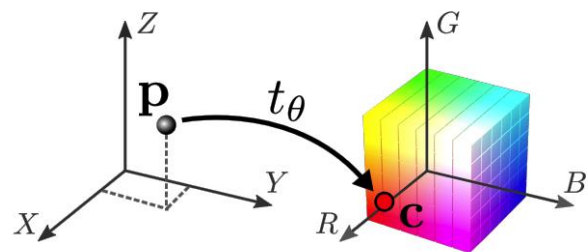


Рис. 2 Схематичне зображення поля текстур

5. Використання полів текстур

На відміну від використовуваних раніше шаблонних методів, метод полів текстур (рис. 2) бере за основу ідею відтворення інформації про текстурі в тривимірному просторі. Параметризація функції відтворення конкретної точки p моделі через глибоку нейронну мережу дає змогу інтегрувати дане представлення в процес глибокого навчання навіть безпосередньо комбінуючи його з

процесом виконання реконструкції текстур об'єктів з можливістю перенесення їх на інші форми об'єктів, спираючись лише на початкову форму об'єкта та його зображення.

6. Формалізація процедури реконструкції

Нехай $t(\cdot)$ – функція, що встановлює відповідність точки тривимірного простору $p \in \mathbb{R}^3$ до кольору $c \in \mathbb{R}^3$, отже відтворює інформацію про текстуру, не змінюючи при цьому представлення даних:

$$t: \mathbb{R}^3 \rightarrow \mathbb{R}^3$$

Параметризуючи функцію t як нейромережа t_θ з параметрами θ , що навчаються, можна спростити завдання до обчислення звичайної регресії. Проте, без використання жодних обмежень може згенерована текстура може не відповідати створеній моделі. Отже, для того, щоб сповістити t_θ про вихідну форму об'єкту було обмежено t_θ значенням форми $s \in S$. Це обмеження веде поле текстур по формі згенерованої моделі та задає перелік точок для виконання передбачення (хмара точок на рис.3). На жаль, навіть при наявності отриманих даних тривимірного об'єкту існує незліченна кількість можливих варіантів для генерації текстур, оскільки сама лише тривимірна модель не визначає ні кольору, ні визначає форми скла чи інших матеріалів, ні освітленості. Однак можна встановити додаткові обмеження про вигляд об'єкту. Обмежуючи t_θ на доданому зображенні з довільного кута зору. Варто зазначити, що зображення не може показати повного вигляду текстури, що буде достатнім для точної реконструкції об'єкта з усіх сторін. Для цього, кодувальник зображення є нейроною мережею та перетворює його представлення на набір глобальних ознак об'єкту $z \in Z$. Таким чином, даний підхід не використовує даних про місце розташування камери, дані освітлення та інші дані навколишнього середовища, чим зумовлює передачу параметрів лише зовнішнього вигляду поверхні моделі.

Підсумовуючи описане вище, отримуємо означення поля текстур[2] як встановлення залежності певної точки моделі p , обмежуючої форми моделі s , та стан об'єкту z до кольору c :

$$t_\theta: \mathbb{R}^3 \times S \times Z \rightarrow \mathbb{R}^3$$

Тобто, за даним визначенням модель здатна передбачити значення кольору c_i для будь-якої точки в просторі p_i .

7. Тренування мережі

Навчання нейронної мережі $t_\theta(p, s, z)$ відбувається з учителем на основі датасету ShapeNET[3]. Нейронна мережа навчається шляхом мінімізації функції втрат, що представляє собою різницю між передбаченим зображенням $\hat{X}$ та реальним зображенням X :

$$\mathcal{L} = \frac{1}{B} \sum_{b=1}^B \sum_{i=1}^{N_b} \|t_\theta(p_{bi}, s_b, z_b) - c_{bi}\|,$$

де:

- B являє собою розмір міні-батчу епохи проходження (кількість оброблюваних фото за один прохід),
- N_b – поверхня пікселів моделі (тобто кількість видимих пікселів моделі),
- c_{bi} – колір відповідної точки реального об'єкту.

Варто зазначити, що закодовані параметри форми s_b та параметри зображення z_b безпосередньо залежать від параметрів форми та об'єкту відповідно та є незмінними незалежно від обраної точки моделі. Тренування відбувається зв'язано для трьох нейромереж (кодувальника форми, кодувальника об'єкту, поля текстур) та значення похибки змінює вагові коефіцієнти кожного з них.

Для виконання експериментального дослідження було використано повнозв'язну архітектуру ResNET[4] для мережі $t_\theta(\cdot, s, z)$, архітектуру ResNet-18[4], переднавчену на датасеті ImageNet для реалізації кодувальника зображень, а також PointNet[5] архітектуру для кодувальника форми. Моделі з учителем пройшли тренування використовуючи оптимізатор Adam з коефіцієнтом швидкості навчання 10^{-4} .

8. Результати досліджень

Для дотримання якісних даних похибки відтворення текстур для їх генерації було використано моделі з датасету ShapeNet для уникнення викривлення похибки генерації текстур через кореляцію з похибкою обрахування форми об'єкту.

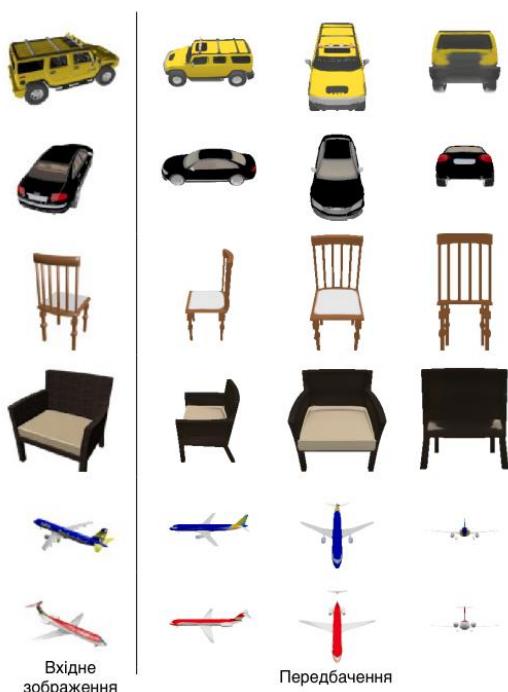


Рис. 4 Результат генерації текстур

Після проходження повного циклу навчання отриманої системи нейронних мереж було проведено валідацію нейромережі, використовуючи зображення та моделі, що не були задіяні в процесі тренування мереж. В процесі валідації даних було згенеровано текстури моделей, проєкції різних сторін яких зображені на рис.4.

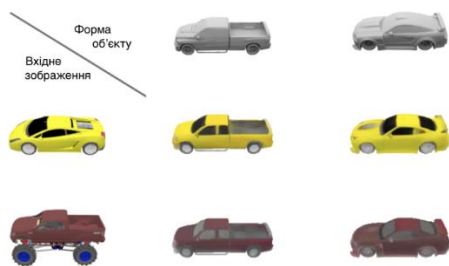


Рис. 5 Генерація текстур на моделі довільної форми

Для демонстрації універсальності отриманої системи, було виконано генерацію текстур на основі зображення при встановленій формі, що не відповідає вхідному зображенню. Відповідно, згенеровані текстури зображені на рис.5. Даний експеримент показує гнучкість системи та можливості кодувальника зображення передавати загальні ознаки об'єкту, а в свою чергу поля текстур мають змогу вдало їх інтегрувати до будь-яких форм.

Інтерпретація отриманих результатів:

Отримані результати демонструють цінність гнучкості полів текстур для генерації

карт текстур тривимірних об'єктів з поєднанням диференційованого рендеру, що додає гнучкості через наявність параметризації, що дозволяє точніше оцінити похибку обчислень, а відповідно – більш точно навчити мережу генерації.

Для оцінки якості генерації текстур використовують декілька метрик, серед яких найбільш вживаними є:

- Початкова відстань Фреше (FID), що використовується для якісної оцінки різниці зображень;
- Метрика структури схожості зображень (SSIM), що виявляє локальні розбіжності зображень;
- Метрика ознак, що базується на використанні нейромережі генерації ознак для порівняння згенерованих характеристик вхідного зображення та зображення після генерації та рендерингу.

FID та SSIM використовуються для точного порівняння результатів роботи генеративної нейромережі генерації зображень та в точності проводить порівняння вхідного зображення з генерованими, тому додатково було введено метрику ознак, що може показати чи передбачення нейромережі генерації ознак буде схожим при генерації ознак вихідного зображення. В табл. 1 наведено середні значення метрик порівняння для кожного з тренуваних класів моделей, а також їх середнє значення.

Табл. 1 Якісне порівняння з використанням метрик FID, SSIM та метрики ознак

Клас	FID	SSIM	Метрика ознак
airplanes	9,236	0,968	0,131
cars	24,271	0,885	0,192
chairs	5,791	0,941	0,124
tables	8,846	0,943	0,123
benches	12,965	0,937	0,133
cabinets	24,251	0,932	0,148
sofas	9,618	0,938	0,135
lamps	16,747	0,960	0,161
vessels	21,481	0,929	0,185
Середнє значення	14,801	0,937	0,149

Отже, згідно метрики FID генеровані текстури мають різну точність залежно від класу зображуваного об'єкту, таким чином

найкращі результати показують моделі класу крісел, за SSIM найбільшу точність мають моделі класу літаків, а за метрикою ознак найменшу похибку розпізнавання мають столи. Найгірші показники за всіма метриками отримали автомобілі, враховуючи

наявність багатьох деталей моделі, отримані дані точності та похибок є закономірними. Середнє значення по всіх категоріях є досить високим та перевищує значення метрик побудови категорій іншим системам та методам.

Висновок

На прикладі задачі виконання реконструкції текстур тривимірних об'єктів на основі єдиного зображення було продемонстровано використання комбінації методів м'якого растеризатору та поля текстур для виконання прогнозування та генерації повного набору текстур моделей, описано переваги та властивості описаних методів, а також продемонстровано їх роботу, результати генерації та проведено аналіз результатів, спираючись на метрики FID, SSIM та метрику ознак.

Метод візуалізації з використанням м'якого растеризатору має значну кількість параметрів, що дозволяють оптимізувати та спростити процедуру візуалізації, що дозволяє вживити даний метод до процедури генерації та забезпечити пряме управління процесом рендерингу нейронною мережею на основі вхідних параметрів зображення. В свою чергу метод полів текстур надає змогу абстрагувати процес представлення моделі та її генерації, що надає мережі представлення про зображуваний об'єкт в загальному, незалежно від положення моделі на зображенні.

Список літератури

1. Shichen Liu, Tianye Li, Weikai Chen, Hao Li. Soft Rasterizer: A Differentiable Renderer for Image-based 3D Reasoning. arXiv.org, 1904.01786, 2019
2. Michael Oechsle, Lars Mescheder, Michael Niemeyer, Thilo Strauss, Andreas Geiger. Texture Fields: Learning Texture Representations in Function Space. arXiv.org, 1905.07259, 2019
3. Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. arXiv.org, 1512.03012, 2015.
4. K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2016
5. C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2017

УДК 004.75

СКЛЕЗЬ Р.О.,
ШИМКОВИЧ В.М.

СИСТЕМА АНАЛІЗУ КОРЕКТНОСТІ ВИКОНАННЯ ГРАФІЧНИХ ТЕСТОВИХ ЗАВДАНЬ

У даній статті розглянуто проблему удосконалення якості тестів як елементів системи автоматичного керування навчальним процесом. Засобом визначення кількісних і якісних параметрів технології навчання виступає контроль як один із невід'ємних компонентів процесу діагностування навчальних досягнень студентів. Розглянуто загальні принципи аналізу порівняння тестових схем для створення подальшої активної системи тестування.

КЛЮЧОВІ СЛОВА *Тести, тестування, система активного тестування, MOODLE, алгоритм порівняння, інформаційна система*

This statistic has identified the problem of improving the quality of the tests as the elements of the system and of the automatic certification by the initial process. In order to determine the most important parameters of the technology, there is a control over one of the unavailable components in