

Завгородній В.В.

Державний університет інфраструктури та технологій

Завгородня Г.А.

Державний університет інфраструктури та технологій

Демченко І.В.

Державний університет інфраструктури та технологій

Крамаренко К.С.

Державний університет інфраструктури та технологій

Шевченко І.О.

Державний університет інфраструктури та технологій

Юрченко А.В.

Державний університет інфраструктури та технологій

МЕТОД СТВОРЕННЯ ШТУЧНИХ ТЕКСТУР ІЗ ЗАДАНИМИ ПАРАМЕТРАМИ

Робота присвячена вирішенню питання генерації текстур із заданими параметрами. Зображення текстури розглядається як набір випадкових полів. Дана стаття розглядає випадкові поля та способи їх опису. Випадкове поле є функцією, значеннями якої є випадкові величини. Якщо передбачається використання значень цієї функції в якості генератора текстури, то на значення поля накладаються додаткові обмеження: по-перше всі випадкові величини поля мають один і той же розподіл, а по-друге коваріація між значеннями функції залежить тільки від вектора, що з'єднує точки, в яких ця функція розглядається, тобто функція є стаціонарною.

У статті висунуто гіпотезу, про те, що сприйняття випадкового поля людиною залежить тільки від значень коваріацій.

В роботі наведено розробку методу, за допомогою якого можна отримати текстуру із заданою коваріаційною функцією. Таке завдання виникає, наприклад, при налагодженні програм розпізнавання об'єктів, суттєвою характеристикою яких є випадкова текстура. Стверджується, що випадкові поля мають властивість стаціонарності, тобто коваріація між двома позиціями залежить тільки від вектора, що сполучає позиції цих точок.

Генерація штучних структур, що описують області, які мають однорідні властивості, є корисною функцією в системах аналізу зображень. В даній статті запропонована модель для генерації штучних текстур з високим паралелізмом. Запропонована модель заснована на рівновазі сил взаємодії між вузлами мережі, та дозволяє генерувати різні текстури зображення з різноманітними характеристиками, такими як ступінь деталізації, нелінійність, хаотичність. Окрім цього досліджується властивість самоорганізації для генерації текстури. Наведені експерименти, які демонструють, що запропонована модель забезпечує успішну генерацію текстур із заданою коваріаційною функцією.

Ключові слова: текстура, кореляція, генератор, коваріаційна функція, випадкові поля.

Постановка проблеми. У комп'ютерній графіці прийнято розуміти під текстурою будь-яке зображення, що застосовується при реалізації сцени. В теорії цифрової обробки зображень текстури – це випадкові поля. Дана стаття розглядає випадкові поля та способи їх опису [1-4]. Випадкове поле є функцією $Fun(x, y)$, значеннями якої є випадкові величини. Якщо передбачається вико-

ристання значень цієї функції в якості генератора текстури, то на значення поля накладаються додаткові обмеження [5-6]:

1. Всі випадкові величини $rand$ поля мають один і той же розподіл.

2. Коваріація між $Fun(x_0, y_0)$ та $Fun(x_1, y_1)$ залежить тільки від вектора, що з'єднує точки (x_0, y_0) та (x_1, y_1) , тобто є стаціонарною.

Важливість виконання зазначеної властивості випадкового поля впливає із наступної гіпотези: сприйняття випадкового поля людиною залежить тільки від значень коваріацій [7-9].

Математичного (статистичного) доказу справедливості цієї гіпотези не існує, але у практичних дослідженнях вона вважається виконаною.

Аналіз останніх досліджень і публікацій. Текстури, що описують області, які мають однорідні властивості, є основною частиною інформації, що міститься в зображенні. Генерація штучної текстури – корисна функція в системах синтезу зображення [10-13]. Не дивлячись на її важливість в обробці зображення, текстури зазвичай описуються без формального визначення і наводяться за допомогою спеціальних моделей (структурної і стохастичної) [14-16], що визначені для спеціалізованих досліджень. Генерація ж являє собою великий інтерес, наприклад для моделювання і аналізу природних текстур, таких як супутникове зображення [17-20] або біомедичне зображення зі стохастичними моделями [21-24], включаючи підходи, що базуються на Марковських випадкових полях [25]. Хоча марковські моделі, як відомо, є корисними в розв'язанні задач текстур зображень [26, 27], виконання їх операцій засновано на моделюванні Монте-Карло і вимагає велику кількість часу, яка може бути надто великою для додатків генерації структур.

Постановка завдання. Проблема полягає у розробці методу, за допомогою якого можна отримати текстуру із заданою коваріаційною функцією. Таке завдання виникає, наприклад, при налагодженні програм розпізнавання об'єктів, суттєвою характеристикою яких є випадкова текстура (ліс, поле, водна поверхня). Обмежене застосування ці текстури мають у комп'ютерній графіці.

Виклад основного матеріалу дослідження.

Генерація стаціонарної текстури. У цій статті використовуються терміни «кореляція» та «коваріація». Різниця між ними полягає у тому, що при обчисленні коваріації не передбачається нормування дисперсій випадкових величин та їх центрування не є обов'язковим.

Відомий наступний простий спосіб створення випадкових полів. Нехай задана матриця M_A , елементи $M_A[x, y] = rand_{x,y}$ якої є некорельованими випадковими величинами з однаковим розподілом і нульовим середнім. Це означає, що математичне очікування визначатиметься:

$EV(M_A[x, y], M_A[x', y']) = 0$, якщо є відмінності хоча б в одному з індексів,

або

$EV(M_A[x, y], M_A[x, y]) = K$, якщо індекси однакові, тобто значення K від індексів не залежить.

Виберемо довільну дійсну матрицю M_B розміру $m \times n$, розмір якої значно менший за розмір матриці M_A і побудуємо матрицю M_F :

$$M_F[i, j] = \sum_{x,y} M_A[x+i, y+j] \cdot M_B[x, y] \quad (1)$$

Знайдемо коваріацію двох позицій у матриці M_F :

$$\begin{aligned} Cov[i, j, i', j'] &= EV(M_F[i, j] \cdot M_F[i', j']) = \\ &= EV(\sum_{x,y} M_A[x+i, y+j] \cdot M_B[x, y] \cdot \sum_{x',y'} M_A[x'+i', y'+j'] \cdot M_B[x', y']) = \\ &= \sum_{x,y,x',y'} M_B[x, y] \cdot M_B[x', y'] \cdot EV(M_A[x+i, y+j] \cdot M_A[x'+i', y'+j']) \end{aligned} \quad (2)$$

В отриманій сумі відмінними від нуля є лише ті доданки, у яких $x+i = x'+i'$ та $y+j = y'+j'$. Це означає, що:

$$\begin{aligned} Cov[i, j, i', j'] &= K \cdot \sum_{x,y} M_B[x, y] \cdot M_B[x+i-i', y+j-j'] = \\ &= D[i-i', j-j'] \end{aligned} \quad (3)$$

З виразу (3) випливає, що отримане поле має властивість стаціонарності – коваріація між двома позиціями залежить тільки від вектора, що сполучає позиції цих точок.

Скрипт для створення текстури згідно з викладеним методом буде мати вигляд:

```
import empty as em
from scipy import signal as sgn
from empty.random import random
from matplotlib import pyplot as plt
```

```
def drawPlot(MAx,MAy,MB):
MA=random((MAx,MAy))-0.5
Cov = sgn.correlate2d(MA,MB,mode='same')
plt.imshow(Cov,cmap='gray')
```

Наведемо приклади текстур, побудованих за допомогою матриць M_{B1} та M_{B2} :

```
MB1 = em.float_([[1,2,4],[-1,4,1],[2,0,3]])
MB2 = em.float_([[ -3,2,1],[1,-5,-1],[-3,0,0]])
```

Відповідно до визначення, значення функції $D(a,b)$ у виразі (3) не залежать від знака аргументу, тому можемо вважати ці аргументи невід'ємними числами. Теоретичні значення цієї функції знаходяться за допомогою наступного скрипту:

```
Koef=1/12
def findD(MB,a,b):
''' Covariation D(a,b)
```

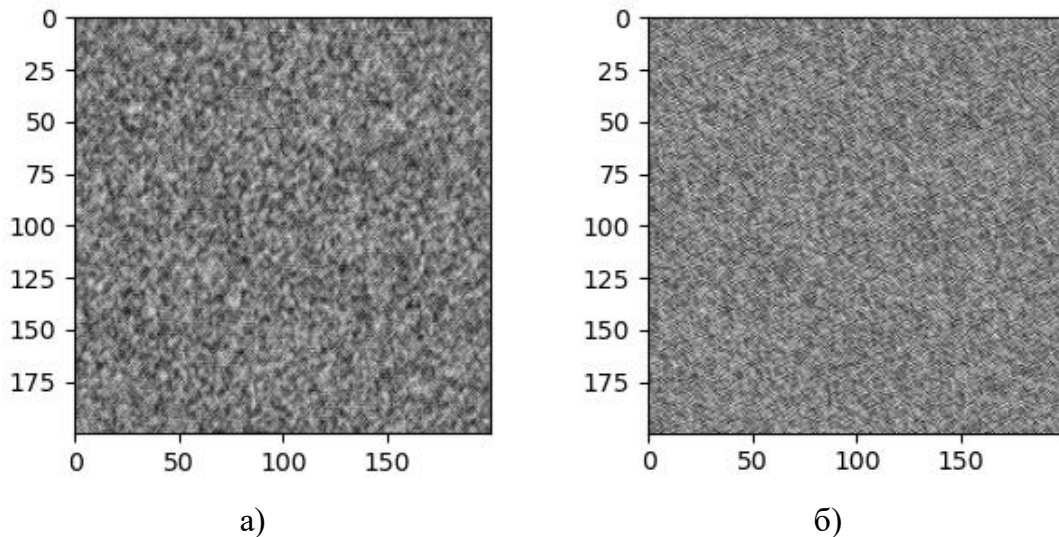


Рис. 1. а) Текстура, побудована за матрицею M_{B1} ; б) Текстура, побудована за матрицею M_{B2}

```
'''
m,n = MB.shape
if a>=m or b>=n:
return 0
```

```
M1 = em.zeros((m+a,n+b))
M1[:m,:n] = MB
M2 = em.zeros((m+a,n+b))
M2[-m,-n:] = MB
return Koef*em.sum(M1*M2)
```

Тут константа $Koef$ – це значення K у виразі (3). Величина $1/12$ знаходиться як дисперсія випадкової величини з рівномірним розподілом на інтервалі $[-0.5,0.5]$. Результати підрахунку коваріацій для зазначених матриць M_B наведені в таблицях 1 та 2.

Порівнюючи дані таблиць 1 і 2 очевидними є відмінності як текстур, так і матриць коваріацій.

Часткове вирішення проблеми створення текстури за матрицею коваріацій. У загальному випадку вирішити рівняння (3) відносно елементів матриці M_B можна лише наближено і не дуже зрозуміло, як це зробити. Розглянемо окремий випадок, коли матриця M_B складається з однакових рядків R довжини $m - R = r[0], r[1], \dots, r[m - 1]$.

У цьому випадку елементи різних рядків матриці M_C не корелюють, тому достатньо розглянути кореляції лише в одному рядку цієї матриці. Рівняння (3) набуває наступного вигляду:

$$D(a) = K \cdot \sum_k r[k]r[k+a] \quad (4)$$

Спосіб відшукання коефіцієнтів $r[k]$ у виразі (4) був запропонований Фейєром понад 100 років тому. На основі рівняння (3):

$$P(rand) = D(m-1) + D(m-2)rand + \dots + D(0)rand^{m-1} + D(1)rand^m + \dots + D(m-1)rand^{2m-2} \quad (5)$$

Тут множник K не грає ніякої ролі. Після цього знаходимо всі корені многочлена $P(rand)$ і залишаємо тільки ті з них, які лежать усередині одиничного кола. Проблеми виникають, якщо існують корені на одиничному колі. По кореням всередині одиничного кола відновлюємо многочлен $Q(rand)$ з точністю до нормування. Коефіцієнти знайденого многочлена після множення на потрібний множник являють собою розв'язання задачі.

Тепер алгоритм розв'язання задачі про наближене створення текстури за заданими значеннями $D(a,b)$ має такий вигляд:

1. За значеннями $D(0,0), D(0,1), \dots, D(0, n-1)$ знаходимо рядок $R1$.

Таблиця 1
Теоретичні значення коваріацій $D(a,b)$.
Матриця M_{B1}

a/b	0	1	2
0	4.33	0.83	0.75
1	1	1.5	-0.17
2	1.17	0.5	0.25

Таблиця 2
Теоретичні значення коваріацій $D(a,b)$.
Матриця M_{B2}

a/b	0	1	2
0	4.17	-0.33	-0.33
1	-1.42	1.08	0.25
2	0.75	0	0

2. За значеннями $D(0,0), D(0,1), \dots, D(0, m-1)$ знаходимо стовпець $R2$.

3. Підраховуємо $m \times n$ матрицю $M_B = R1 \cdot R2$ та застосовуємо викладений вище спосіб генерації текстури.

Застосуємо цей спосіб до елементів таблиці 1.

Многочлен $P(rand)$ має коефіцієнти 0.75, 0.84, 4.33, 0.84, 0.75. Знаходимо корені цього многочлену за допомогою функції *empty.roots*, два з них лежать усередині одиничного кола. За допомогою функції *em.poly* знаходимо коефіцієнти многочлена $Q(rand)$. Ці коефіцієнти множаться на нормуючий множник для того, щоб добуток старшого та молодшого коефіцієнтів дорівнював 0.75. В результаті отримаємо вектор $R1 = 2.02, 0.35, 0.37$ (розташування коефіцієнтів за зростанням або зменшенням ступенів значення не має). Аналогічно знайдемо вектор $R2 = 1.95, 0.39, 0.60$, використовуючи перший стовпець таблиці 1.

Текстура, породжена цими векторами, представлена на рисунку 2.

Розглянувши текстуру на рисунку 2 можна побачити, що зовні вона не відрізняється від оригіналу, представленого на рисунку 1а.

Висновки. Генерація штучних структур, що описують області, які мають однорідні властивості, є корисною функцією в системах аналізу

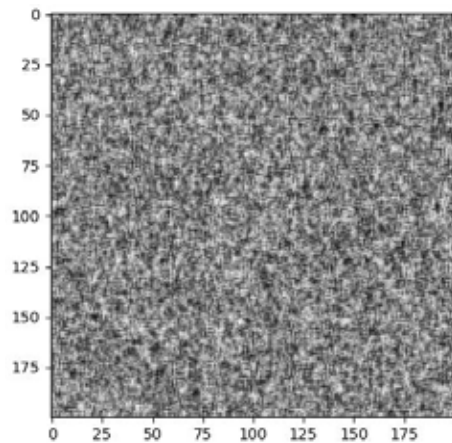


Рис. 2. Текстура, побудована за одновимірними векторами

зображень. В даній статті запропонована модель для генерації штучних текстур з високим паралелізмом. Запропонована модель заснована на рівновазі сил взаємодії між вузлами мережі, та дозволяє генерувати різні текстури зображення з різноманітними характеристиками, такими як ступінь деталізації, нелінійність, хаотичність. Окрім цього досліджується властивість самоорганізації для генерації текстури. Наведені експерименти, які демонструють, що запропонована модель забезпечує успішну генерацію текстур із заданою коваріаційною функцією.

Список літератури:

1. P. Arbelaez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. CVPR, 2014.
2. S. Bell, P. Upchurch, N. Snavely, and K. Bala. Opensurfaces: A richly annotated catalog of surface appearance. In Proc. SIGGRAPH, 2013.
3. K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In Proc. BMVC, 2014.
4. M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi. Describing textures in the wild. In Proc. CVPR, 2014.
5. Dodonov, A., Mukhin, V., Zavgorodnii, V., Kornaga, Ya., Zavgorodnya A. (2021). Method of searching for information objects in unified information space. System research and information technologies. No 1(2021). P. 34-46. DOI : <https://doi.org/10.20535/SRIT.2308-8893.2021.1.03>
6. Дранишников Л.В., Загородний В.В. Анализ и оценка риска возникновения техногенных аварий с целью управления их безопасностью на основе информационных технологий. Нові технології». 2008. No4 (22). С. 119-129.
7. R. B. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proc. CVPR, 2014.
8. Y. Gong, L. Wang, R. Guo, and S. Lazebnik. Multi-scale orderless pooling of deep convolutional activation features. In Proc. ECCV, 2014.
9. B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In Computer Vision–ECCV 2014, pages 297-312. Springer, 2014.
10. K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In Proc. ECCV, 2014.
11. P. Isola, D. Zoran, D. Krishnan, and E. H. Adelson. Crisp boundary detection using pointwise mutual information. In Proc. ECCV, 2014.

12. A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In Proc. NIPS, 2012.
13. Zavgorodnii V., Mukhin V., Komaga Y., Zavgorodnya A., Herasymenko O., Mukhin O. Social Risk Assessment Mechanism Based on the Neural Networks. International Conference on Advanced Trends in Information Theory (ATIT-2019). 18-20 December 2019. P. 179-182. DOI: <https://doi.org/10.1109/ATIT49449.2019.9030519>
14. L. Liu, L. Wang, and X. Liu. In defense of soft-assignment coding. In Computer Vision (ICCV), 2011 IEEE International Conference on, pages 2486–2493. IEEE, 2011.
15. M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. In Proc. CVPR, 2014.
16. O. M. Parkhi, K. Simonyan, A. Vedaldi, and A. Zisserman. A compact and discriminative face track descriptor. In Proc. CVPR, 2014.
17. X. Qi, R. Xiao, C. G. Li, Y. Qiao, J. Guo, and X. Tang. Pairwise rotation invariant co-occurrence local binary pattern. PAMI, 36(11): 2199-2213, Nov 2014.
18. Завгородній В.В., Завгородня Г.А. Метод подання знань про оцінку ризику виникнення техногенних аварій. Вісник Кременчуцького національного університету імені Михайла Остроградського, 2018. Випуск № 4 (111). С. 43-48. DOI: <https://doi.org/10.30929/1995-0519.2018.4.43-48>
19. A. S. Razavin, H. Azizpour, J. Sullivan, and S. Carlsson. Cnn features off-the-shelf: An astounding baseline for recognition. In DeepVision workshop, 2014.
20. Завгородній В.В., Отрох С.І., Завгородня Г.А., Грищенко О.О. Аналіз методів подання знань при розпізнаванні надзвичайних ситуацій техногенного характеру. Наукові записки Українського науково-дослідного інституту зв'язку, 2018. № 3. С. 59-69.
21. G. Schwartz and K. Nishino. Visual material traits: Recognizing per-pixel material context. In Proc. CVCP, 2013.
22. L. Sharan, C. Liu, R. Rosenholtz, and E. H. Adelson. Recognizing materials using perceptually inspired features. International Journal of Computer Vision, 103(3): 348-371, 2013.
23. M. Sulc and J. Matas. Fast features invariant to rotation and scale of texture. Technical report, 2014.
24. Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan. Cnn: Single-label to multi-label. 2014.
25. Zavgorodnii V., Zavgorodnya A., Maiko V., Malikov V., Zhuk D. Methods And Models For Assessment Of Reliability Of Structural-Complex Systems. World Science, 2018. № 11 (39). P. 5-14. DOI: https://doi.org/10.31435/rsglobal_ws/30112018/6227
26. N. Zhang, J. Donahue, R. Girshick, and T. Darrell. Part-based RCNNs for fine-grained category detection. In Proc. ECCV, 2014.
27. B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In Proc. NIPS, 2014.

Zavgorodnii V.V., Zavgorodnya A.A., Demchenko I.V., Kramarenko K.S., Shevchenko I.O., Yurchenko A.V. METHOD OF CREATING ARTIFICIAL TEXTURES WITH SPECIFIED PARAMETERS

The work is devoted to solving the problem of generating textures with given parameters. The texture image is considered as a set of random fields. This article discusses random fields and ways to describe them. A random field is a function whose values are random variables. If the values of this function are to be used as a texture generator, additional restrictions are imposed on the field values: first, all random field values have the same distribution, and second, the covariance between the values of the function depends only on the connecting vector: points at which this function is considered, ie the function is stationary.

The article hypothesizes that human perception of a random field depends only on the values of covariances.

The paper presents the development of a method by which a texture with a given covariance function can be obtained. This problem arises, for example, when setting up object recognition programs, an essential characteristic of which is a random texture. It is argued that random fields have the property of stationarity, ie the covariance between two positions depends only on the vector connecting the positions of these points.

Generation of artificial structures describing areas that have homogeneous properties is a useful function in image analysis systems. This paper proposes a model for generating artificial textures with high parallelism. The proposed model is based on the balance of forces of interaction between network nodes, and allows to generate different image textures with different characteristics, such as the degree of detail, nonlinearity, chaos. In addition, the property of self-organization for texture generation is investigated. Experiments are presented that demonstrate that the proposed model provides successful generation of textures with a given covariance function.

Key words: texture, correlation, generator, covariance function, random fields.