

ЗАСТОСУВАННЯ МОДЕЛЕЙ ГЛИБОКОГО НАВЧАННЯ ДЛЯ ВИДАЛЕННЯ ШУМІВ НА ЗОБРАЖЕННЯХ

Максим Гавриленко

ORCID: <https://orcid.org/0009-0000-4339-0254>

Дніпропетровський університет ім. Олеся Гончара, Дніпро, Україна

Ольга Мацуга

ORCID: <https://orcid.org/0000-0001-6444-8566>

Дніпропетровський університет ім. Олеся Гончара, Дніпро, Україна

Вступ

У сучасному цифровому світі якість зображень відіграє важливу роль у різних сферах, включаючи медицину, науку, розваги та комунікацію. Шум, що виникає внаслідок недосконалості обладнання, проблем в передавальних каналах або випадкових фізичних факторів, значно знижує ефективність обробки та сприйняття зображень.

Традиційні методи фільтрації, такі як гаусівський, медіанний, білатеральний або нелінійний адаптивний фільтри, є ефективними, проте мають свої обмеження, особливо в умовах складних шумових характеристик [1; 2]. Натомість, глибокі нейронні мережі демонструють високу ефективність у задачах приглушення шуму завдяки здатності навчатися на великих обсягах даних і виявляти складні патерни шуму [3–5]. Актуальною при цьому є задача вибору найкращої архітектури нейронної мережі для видалення шуму із зображень.

ЦІЛЬ ТА ЗАДАЧІ

Ціллю даної роботи було дослідити різні архітектури нейронних мереж в задачі видалення шумів на зображеннях. Для досягнення цілі потрібно було виконати наступні завдання:

1. Створити набір даних, що містить оригінальні зображення разом з їх зашумленими версіями.
2. Обрати і навчити декілька моделей глибокого навчання для видалення шумів із зображень.
3. Провести порівняльний аналіз роботи навчених моделей та класичних фільтрів.

МАТЕРІАЛИ ТА МЕТОДИ

Для навчання і тестування моделей нейронних мереж потрібно

мати відповідний набір зображень. В роботі було створено навчальну, валідаційну та тестову вибірки, які містили оригінальні зображення та зображення з доданим шумом. Для їх формування було використано 3 набори зображень з www.kaggle.com: 1) Butterfly Image Classification – містить зображення різних метеликів [6]; 2) 100 Sports Classification – зображення спортсменів [7]; 3) UTKFace – зображення обличч людей від 0 до 116 років [8].

Навчальна вибірка формувалася у такий спосіб. З кожного з трьох наборів випадково було обрано 1000 зображень і вони масштабувалися до розміру 128x128. Потім з цих зображень було випадково обрано 30, до кожного з яких вносився певний тип шуму різного ступеня (всього було 10 ступенів шуму). Тим самим, отримувались 300 зображень з певним типом шуму різного ступеня. І так робилося для кожного з 6 видів шуму. У підсумку навчальна вибірка містила 1800 зображень. Валідаційна вибірка формувалася так само, лише її розмір був в 2 рази менший. Тестова вибірка формувалася на основі 100 оригінальних зображень із додаванням до них шуму трьох ступенів.

В процесі формування вибірок до зображень додавався шум одного з шести видів: гаусівський, Пуассона, «сіль та перець», Лапласа, змішаний, квантування (рис. 1.1) [1; 2; 9].

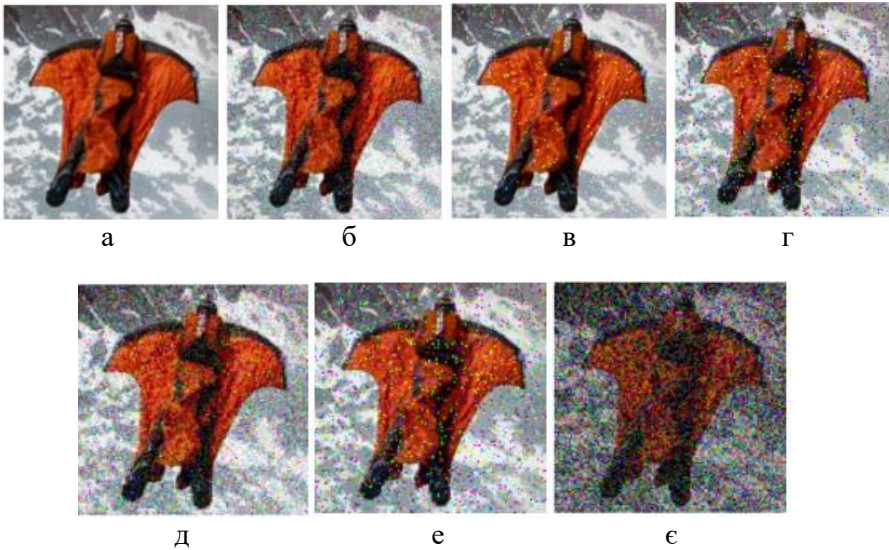


Рисунок 1 – Оригінальне зображення (а) та зашумлені зображення: б – гаусівський шум, в – шум Пуассона, г – шум «сіль і перець», д – шум Лапласа, е – змішаний шум, є – шум квантування

В якості моделей глибоких нейронних мереж для видалення шуму із зображень було розглянуто згортковий автоенкодер та U-Net. Згортковий автоенкодер є досить широкоживаною моделлю для видалення шумів із зображень [3–5, 10]. U-Net розроблена для задачі сегментації зображень. Проте в роботі [11] була спроба її застосування для видалення шуму із зображень. В роботі було навчено 3 моделі:

1) модель згорткового автоенкодера, яка містила 3 згорткові шари в енкодері та 3 в декодері;

2) ще одна модель згорткового автоенкодера, яка була глибшою і містила 6 згорткових шарів в енкодері та декодері;

3) модель U-Net, в якій було змінено вихід та функцію втрат.

В якості функції втрат використовувалася функція MSE (Mean Squared Error). Як метрики якості були взяті PSNR (пікове співвідношення сигнал-шум) і SSIM (індекс структурної схожості).

Навчання проводилося в середовищі Google Colaboratory (Colab), яке надає відмінну можливість працювати з Jupyter Notebook онлайн, запускаючи код у хмарі Google. Використання T4 GPU в Colab дозволило значно прискорити обчислення.

В ході навчання використовувалася навчальна вибірка з 1800 зображень і валідаційна з 900 зображень. Як метод оптимізації був застосований Adam. За результатами експериментів початкову швидкість навчання було обрано 0.000003 для U-Net, 0.000009 для першого автоенкодера, 0.000025 для другого. Dropout не застосовувався.

В ході навчання усіх моделей спостерігалася проблема вибухання градієнтів. Для її усунення використовувався метод градієнтного обрізання (gradient clipping) за допомогою GradientTape з TensorFlow. Цей інструмент дозволяє автоматично обчислювати градієнти для змінних відносно певного об'єкта `tf.Variable` під час проходження уперед через модель. Результати навчання для моделі U-Net наведено на рисунку 3.

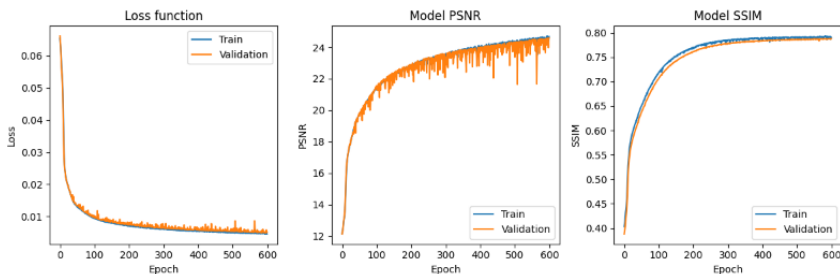


Рисунок 3 – Результати навчання моделі U-Net протягом 600 епох

РЕЗУЛЬТАТИ

З метою порівняти роботу навчених моделей між собою, а також з класичними фільтрами, було виконано наступне. Було взято 100 тестових зображень і кожне зашумлено певним видом шуму з різним ступенем зашумлення. Всього було розглянуто 3 ступені зашумлення. Після цього шум видалявся за допомогою трьох навчених моделей та класичних фільтрів і розраховувалися метрики PSNR та SSIM. Далі метрики усереднювалися за 100 зображеннями. Під час застосування класичних фільтрів для першого ступеня зашумлення брався фільтр з ядром 3×3 , для другого – з ядром 5×5 , і третього – з ядром 7×7 (лише для у випадку шуму Лапласа 9×9). Ці розміри фільтрів були визначені як оптимальні в ході експериментів.

Результати порівняння показали, що U-Net в більшості випадків забезпечувала кращу якість видалення шуму, ніж згорткові автоенкодера. Тому нижче подані і проаналізовані результати лише для моделі U-Net.

Усереднені результати експерименту з порівняння роботи U-Net та класичних фільтрів подано в таблиці 1.

На зображеннях з невеликою кількістю гаусівського шуму нейронна мережа майже досягає результатів фільтрів. При високому ступені зашумленості за PSNR перевагу має нейронна мережа.

На зображеннях шумом Пуассона нейронна мережа показала гірші результати, ніж медіанний фільтр. Проте чим вищою був ступінь зашумлення, тим різниця в якості між нейронною мережею і класичним фільтром була меншою.

На зображеннях з невеликою кількістю шуму «сіль та перець» нейронна мережа трохи поступається медіанному фільтру. При середньому ступені зашумленості нейронна мережа справляється краще, але при найвищій ступені зашумлення фільтри є кращим рішенням.

На зображеннях з невеликою кількістю шуму Лапласа нейронна мережа спроможна видаляти шум майже як фільтри, в деяких випадках навіть краще.

На зображеннях з невеликою кількістю змішаного шуму нейронна мережа показує трохи гірші результати за медіанний фільтр, але не сильно йому поступається.

На зображеннях з невеликою кількістю шуму квантування нейронна мережа впоралась краще фільтрів. Однак при високому ступені зашумленості нейронна мережа трохи відстає за результатами метрик від фільтрів.

Таблиця 1 – Усереднені значення метрик якості моделі U-Net та класичних фільтрів на тестових даних

	Ступінь зашумлення		
	1	2	3
Для зображень з гаусівським шумом			
Нейронна мережа U-Net	PSNR:27.15 SSIM:0.90	PSNR:21.23 SSIM:0.65	PSNR:18.22 SSIM:0.51
Фільтр Гауса	PSNR:27.47 SSIM:0.84	PSNR:19.80 SSIM:0.60	PSNR:17.62 SSIM:0.6
Для зображень з шумом Пуассона			
Нейронна мережа U-Net	PSNR:28.36 SSIM:0.90	PSNR:23.20 SSIM:0.85	PSNR:20.61 SSIM:0.81
Медіанний фільтр	PSNR:34.88 SSIM:0.96	PSNR:27.72 SSIM:0.90	PSNR:23.97 SSIM:0.82
Для зображень з шумом «сіль та перець»			
Нейронна мережа U-Net	PSNR:25.62 SSIM:0.848	PSNR:29.79 SSIM:0.61	PSNR:11.86 SSIM:0.37
Медіанний фільтр	PSNR:33.31 SSIM:0.96	PSNR:26.73 SSIM:0.88	PSNR:14.54 SSIM:0.51
Для зображень з шумом Лапласа			
Нейронна мережа U-Net	PSNR:26.77 SSIM:0.89	PSNR:20.40 SSIM:0.60	PSNR:17.76 SSIM:0.50
Медіанний фільтр	PSNR:34.79 SSIM:0.96	PSNR:19.57 SSIM:0.49	PSNR:18.83 SSIM:0.48
Для зображень зі змішаним шумом			
Нейронна мережа U-Net	PSNR:28.01 SSIM:0.94	PSNR:23.92 SSIM:0.79	PSNR:22.86 SSIM:0.73
Медіанний фільтр	PSNR:33.34 SSIM:0.95	PSNR:29.91 SSIM:0.91	PSNR:27.28 SSIM:0.87
Для зображень з шумом квантування			
Нейронна мережа U-Net	PSNR:28.22 SSIM:0.93	PSNR:25.79 SSIM:0.86	PSNR:24.94 SSIM:0.84
Медіанний фільтр	PSNR:26.69 SSIM:0.85	PSNR:26.69 SSIM:0.85	PSNR:26.8 SSIM:0.85

Висновки

У роботі розглянуто задачу видалення шумів із зображень за допомогою нейронних мереж, а також представлено експериментальні результати тренування і тестування моделей нейронних мереж на зображеннях зі штучно доданим шумом. Було створено набір кольорових зображень розміром 128x128 з шістьма видами шумів з різними ступенями зашумлення. На цьому наборі навчено два згорткові атоенкодеери та модель U-Net для видалення шумів.

Роботу навчених моделей було порівняно між собою та з

класичними фільтрами на тестових зображеннях. Результати експериментів засвідчили, що модель U-Net краще справляється з видалення шуму із зображень, ніж автоенкодер. У порівнянні з класичними фільтрами модель U-Net демонструє конкурентоспроможні результати. Вона показує кращу якість у випадку шумів гаусівського, Лапласа та квантування, особливо в контексті збереження структурної інформації (вищі значення SSIM).

ПОСИЛАННЯ

1. Gonzalez, R., & Woods, R. (2017). *Digital Image Processing*. <https://dl.icdst.org/pdfs/files4/01c56e081202b62bd7d3b4f8545775fb.pdf>
2. Szeliski, R. (2022). *Computer Vision: Algorithms and Applications* (2nd ed). Springer. <https://szeliski.org/Book>
3. Elad, M., Kowar, B., & Vaksman, G. (2023). Image Denoising: The Deep Learning Revolution and Beyond – A Survey Paper. *SIAM Journal on Imaging Sciences*. 16(3). 1594-1654. <https://doi.org/10.1137/23M1545859>
4. Chintan, D. (2021, January 23). Image Denoising Using Deep Learning. *Medium*. <https://pub.towardsai.net/image-de-noising-using-deep-learning-1a8334c81f06>
5. Solomon, S. (2021, August 27). Image Denoising using Deep Learning. *Medium*. <https://medium.com/analytics-vidhya/image-denoising-using-deep-learning-dc2b19a3fd54>
6. *Butterfly Image Classification*. (n.d.). Kaggle. <https://www.kaggle.com/datasets/phuchthai02/butterfly-image-classification>
7. *100 Sports Image Classification*. (n.d.). Kaggle. <https://www.kaggle.com/datasets/gpiosenka/sports-classification/data>
8. *UTKFace*. (n.d.). Kaggle: <https://www.kaggle.com/datasets/jangedoo/utkface-new>
9. Shrivastava, A., Bisen, P., Dubey, M., & Choudhari, M. (2015). Image Denoising Using Different Filters (A Comparison of Filters). *International Journal of Emerging Trends in Science and Technology*. 2(4). 2214-2219. <https://journals.indexcopernicus.com/api/file/viewByFileId/174827.pdf>
10. Kuo, C. (2019, November 20). Convolutional Autoencoders for Image Noise Reduction. *Medium*. <https://towardsdatascience.com/convolutional-autoencoders-for-image-noise-reduction-32fce9fc1763>
11. Paavani, D. (2019). Image Denoising Using a U-net. https://stanford.edu/class/ee367/Winter2019/dua_report.pdf