Гриценко В. В., Бушин І. М.
Черкаський національний університет ім. Б. Хмельницького

ВІДНОВЛЕННЯ ЗАШУМЛЕНИХ ЦИФРОВИХ ЗОБРАЖЕНЬ

Відновлення різкості зображення зі значними деталями з вхідного зашумленого зображення, вже давно є активною областю досліджень у сфері комп'ютерного зору. Зі збільшенням використання портативних пристроїв, таких як мобільні телефони та бортові камери, зашумленість стає усюдисущою проблемою, з якою потрібно боротися.

На практиці зображення часто спотворюються шумом, який з'являється на етапах їхнього створення та/або передачі. У наукових публікаціях значну увагу приділено визначенню різних видів шумів. На підставі аналізу літературних джерел встановлено, що шум є накладеною на зображення маскою пікселів випадкового кольору та яскравості. Причинами виникнення шуму на зображенні можуть бути несправність використовуваної апаратури, збої в роботі каналу зв'язку, несприятливі зовнішні умови та ін. Шуми, що виникають при цьому, підлягають класифікації з метою їх вивчення, формалізації для подальшого усунення чи мінімізації їх шкідливого впливу. Існує досить багато типів шумів, які негативно впливають на оброблення та аналіз зображень. Серед них – гаусівський шум, шум Пуассона, імпульсний шум, шум зерен фотоплівки, спекл-шум, шум Перліна та ін.

Фундаментальним завданням в області обробки зображень є ефективне видалення шуму, тобто побудова деякого наближення вихідного зображення по заданому (зашумленому) зображенню. Очевидно, що наше наближення має бути якомога ближче до вихідного зображення. Складність вирішення даного завдання істотно залежить від розглянутої моделі шуму. В принципі, чим більше ми знаємо про оператор спотворення і про функції шуму, тим ближче буде оброблене зображення до вихідного зображення [1, 5-14].

Найчастіше використовуваними техніками шумозаглушення є класичні методи базовані на фільтрах, такі як адаптивна фільтрація, медіанна фільтрація, математична морфологія, розмивання Гауса, вейвлет-перетворення а також фільтри Вінера та ін.

Опріч методів, заснованих на фільтрах, існують підходи до моделювання шуму, що не залежать від навчання, такі як EPLL, KSVD, BM3D, Марковські випадкові поля та Total Variation Denoising (загальна зміна шумоподавлення). Такі методи засновані на попередньому моделюванні шуму, і вони мають деякі недоліки, такі як обчислювальна складність та необхідність точного налаштування параметрів. Їх ефективність сильно залежить від попередніх знань про тип шуму (наприклад, гаусовий, «сіль та перець», тощо) та його статистичні властивості (наприклад, середнє значення та дисперсія).

На сьогодні обчислювальна техніка дуже добре розвинена, також добре опрацьовані алгоритми навчання глибоких нейронних мереж, які можливо застосовувати не тільки в задачах розпізнавання та класифікації. Останнім часом з розвитком технологій глибокого навчання з'явилася можливість реалізовувати також різні методи видалення шуму з зображень, що показують високі результати [2].

Методи глибокого навчання стали найефективнішими методами, що використовуються в багатьох реальних проблемах цифрової обробки зображень.

Як можна дізнатися з відповідної наукової літератури, алгоритми на основі нейронних мереж нічим не поступаються «ручним» просторовим алгоритмам, а в багатьох випадках перевершують їх за якістю. Вони використовуються як варіант заміни незалежних від навчання фільтрів, та підходів, що потребують попередніх знань про наявні шуми.

Перевагою таких підходів є те, що нема потреби підбирати особливе математичне перетворення для вирішення завдання шумозаглушення, нейронні мережі роблять це самі. Також, на відміну від колишніх технологій, нейромережі дають вищу якість вихідних зображень, зокрема, зменшують «замиленість» зображення і як правило, менше піддаються впливу нелінійних характеристик механізмів генерації шуму.

Серед таких підходів багатошарові перцептрони (MLP) протягом тривалого часу були одним з найбільш досліджуваних методів машинного навчання для відновлення зашумлених

зображень [3, 2392-2399], [4, 1067–1074], [5, 390-396]. З недавніми зростанням потужності обробки комп'ютерної графіки MLP замінили згорткові нейронні мережі (CNN), особливо стосовно завдань обробки зображень.

Згорткова нейромережа (ЗНМ) має спеціальну архітектуру, яка дозволяє їй максимально ефективно аналізувати образи. Сама ідея ЗНМ базується на чередуванні згорткових та субдискретизуючих шарів (pooling), а її структура є однонаправленою. ЗНМ отримала свою назву від операції згортки, яка передбачає, що кожен фрагмент зображення буде помножено на ядро згортки поелементно, при цьому отриманий результат повинен бути підсумований та записаний у схожу позицію вихідного зображення. Така архітектура забезпечує інваріантність розпізнавання відносно зсуву об'єкта, поступово збільшуючи «вікно», на яке «дивиться» згортка, виявляючи все більші й більші структури і шаблони у зображенні.

Розглянемо моделі процесів спотворення та відновлення зображень. Як показано на рис. 1, процес спотворення зображення полягає в тому, що деякий оператор спотворення *H* діє на вихідне зображення *f(x, y)*, так, що в результаті додавання адаптивного шуму отримуємо спотворене зображення *g(x, y)*.

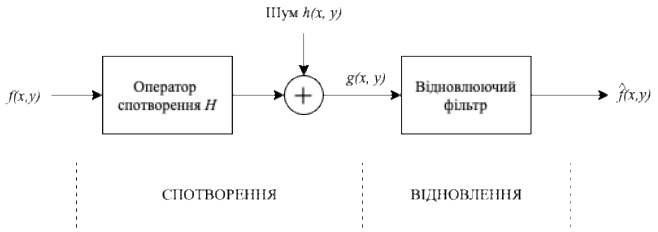

Рис. 1 – Модель процесу спотворення/відновлення зображень

Процес відновлення полягає в побудові деякого наближення вихідного зображення по спотвореному зображенню *g(x, y)*, за деякою інформацією щодо оператора спотворення *H* і адитивного шуму *η(x, y)*. При цьому наше наближення має бути якомога ближче до вихідного зображення. Спотворене зображення може бути представлене в просторовій області у вигляді (1).

$$\boldsymbol{g(x,y) = h(x,y) * f(x,y) + \eta(x,y),} \qquad \textbf{(1)}$$

де *h(x, y)* – функція, що представляє спотворює оператор в просторовій області, а символом «*» позначається згортка.

Згортка в просторовій області еквівалентна множенню в частотній області, тому вище наведена рівність (1) може бути представлена в частотної області у такому вигляді:

$$\boldsymbol{G(u,v) = H(u,v)F(u,v) + N(u,v),} \qquad \textbf{(2)}$$

де зазначені функції – Фур'є-образи відповідних їм функцій в (1).

Так як спотворення зображення це результат згортки, то для його відновлення необхідно знайти такий фільтр, застосування якого призводило б до зворотного процесу. Тому для позначення лінійного процесу відновлення часто використовується термін реконструкція (деконволюція) зображень. Аналогічно, фільтри для відновлення часто називаються реконструюючими фільтрами.

Нами передбачається розробка алгоритму видалення шумів на цифрових зображеннях за допомогою ЗНМ та здійснення його програмної реалізації засобами Python з використанням фреймворка для глибокого навчання Keras. Даний фреймворк є надбудовою над TensorFlow – відкритою платформою для машинного навчання, що має гнучку екосистему інструментів, бібліотек та ресурсів, та дозволяє дослідникам просувати найсучасніші технології машинного навчання, а розробники можуть легко розгортати додатки.

Список літератури

1. J. Najeer Ahamed, V. Rajamani Design of hybrid filter for denoising images using fuzzy network and edge detecting // American Journal of Scientific Research, Issue 3, 2009, pp 5-14.
2. Методы удаления шумов на изображениях на основе применения искусственных нейронных сетей [Електронний ресурс] URL: https://www.lib.tpu.ru/fulltext/c/2010/C01/V2/170.pdf
3. H. C. Burger, C. J. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with BM3D?" in Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012, pp. 2392–2399.
4. C. J. Schuler, H. C. Burger, S. Harmeling, and B. Scho ̈lkopf, "A machine learning approach for non-blind image deconvolution," in Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on. IEEE, 2013, pp. 1067–1074.
5. R. G. Pires, D. F. S. Santos, L. A. M. Pereira, G. B. De Souza, A. L. M. Levada, and J. P. Papa, "A robust restricted boltzmann machine for binary image denoising," in 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI). Ieee, 2017, pp. 390–396.

Дмитрюк В., Липовецький О.
Черкаський національний університет ім. Б. Хмельницького

РОЗРОБКА СИСТЕМИ ДЕТЕКЦІЇ РУХУ

Метою роботи є аналіз та вибір технології і створення програмного забезпечення віддаленого відеоспостереження із виявленням руху для системи на основі смартфонів, а також проведення експериментального дослідження створеної системи.

Смартфон з операційною системою Android можна перетворити на IP-камеру для відеоспостереження за об'єктами за допомогою Wi-Fi-з'єднання. IP-камеру можна придбати в магазині за $ 20-50, але, перетворивши свій пристрій Android на IP-камеру, користувач зможе завантажити її безкоштовно.

IP-камера на базі смартфона може використовуватися як камера спостереження. Для перетворення смартфона на IP-камеру, використовується мобільний пристрій Android, програмне забезпечення та доступ до Wi-Fi. IP-камера працює, коли смартфон, на якому встановлена програма, підключений до мережі Wi-Fi або до мобільного оператора. Після підключення IP-камери та початку роботи, користувач зможе отримати до неї доступ через Telegram.

При реалізації проекту створення відеоспостереження за допомогою IP-камери зі звичайного смартфона виникає питання про цілеспрямованість самої ідеї реалізації. З одного боку, є безкоштовний телефон, який шкода викидати, оскільки, з іншого боку, на цей пристрій існують обмеження, які можуть спричинити негативний сценарій використання.

На рис. 1 наведено, запропонований нами алгоритм роботи програми. Клас Reader відповідає за зчитування кадрів із відеопотоку. Після прочитання кадру клас Detection відповідає за визначення та відстеження об'єктів. Обробка зображень, фільтрація, створення фонової моделі, визначення об'єктів виконуються за допомогою методу detectObjects, який містить масив об'єктів. Кожен об'єкт відеоспостереження є екземпляром класу Object. Клас Object містить поля, що зберігають інформацію про об'єкт: скільки кадрів пройшло з моменту першого визначення, кількість кадрів, в яких програма фіксує переміщення об'єкта. Для кожного об'єкта створюється фільтр Калмана, який є екземпляром класу SingleKalmanFilter.

Максимальний розмір зображення – 1920 пікселі в довжину і 1080 в ширину. Послідовність містить об'єкт, який йде по криволінійній траєкторії. Протягом усієї послідовності програма надає можливість бачити зміни в становищі об'єкта, що контролюється. Отже, незважаючи на невеликі розміри об'єкта та спотворення реальної камери, результат цього прикладу повністю виправдовує очікування.

У результаті роботи була розпочата розробка програми для визначення та утримання об'єктів під час відеоспостереження (рис.2). Проаналізовано методи та технології ідентифікації об'єктів. Програмні тести проводились на трьох прикладах, які показали переваги та недоліки розробленого алгоритму.