

Використання глибокого навчання для обернених задач

Середович Віктор

Львівський національний університет імені Івана Франка
Факультет прикладної математики та інформатики

10 травня 2021 р.

Зміст

Постановка задачі

Структура обернених задач

Генерація шуму та оцінка пошкоджень

Модель для автоенкодера

Автоенкодер для розв'язування обернених задач

- Автоенкодер

- Автоенкодер для видалення шуму

Аналіз результатів

Постановка задачі

Оберненими задачами будемо вважати такі задачі, в яких невідомим є n – піксельне зображення $\mathbf{x} \in \mathbb{R}^n$ яке було отримане з m вимірювань $\mathbf{y} \in \mathbb{R}^m$ відповідно до рівняння 1.1.

$$\mathbf{y} = \mathcal{A}(\mathbf{x}) + \epsilon \quad (1.1)$$

де \mathcal{A} - це прямий оператор вимірювання та ϵ є певним вектором шуму.

Якщо розподіл шуму відомий, \mathbf{x} можна відновити розв'язавши задачу оцінки максимальної ймовірності (maximum likelihood) 2.1.

$$\hat{\mathbf{x}}_{\text{ML}} = \arg \max_{\mathbf{x}} p(\mathbf{y}|\mathbf{x}) = \arg \min_{\mathbf{x}} -\log p(\mathbf{y}|\mathbf{x}) \quad (2.1)$$

де $p(\mathbf{y} | \mathbf{x})$ це ймовірність спостереження \mathbf{y} за умови якщо \mathbf{x} є справжнім зображенням.

В залежності від умов задачі, можуть бути відомі попередні дані про те яким має бути \mathbf{x} . Ці умови можна використати для формулювання задачі оцінки максимальної апостеріорної ймовірності (maximum a posteriori), що приводить до задачі 2.2.

$$\hat{\mathbf{x}}_{\text{MAP}} = \arg \max_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}) = \arg \max_{\mathbf{x}} p(\mathbf{y}|\mathbf{x})p(\mathbf{x}) = \arg \min_{\mathbf{x}} -\ln p(\mathbf{y}|\mathbf{x}) - \ln p(\mathbf{x}) \quad (2.2)$$

Для випадку білого гаусівського шуму, цільову функцію можна сформулювати як:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \frac{1}{2} \|\mathcal{A}(\mathbf{x}) - \mathbf{y}\|_2^2 + \lambda R(\mathbf{x}) \quad (2.3)$$

Більш об'єктивною альтернативою є SSIM (structural similarity index measure) метрика яка була представлена в роботі.

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (3.1)$$

де

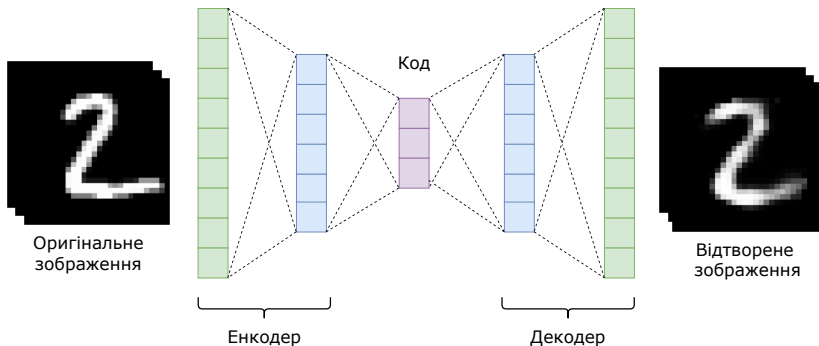
- ▶ μ_x, μ_y середні значення x, y
- ▶ σ_x^2, σ_y^2 дисперсія x, y
- ▶ σ_{xy} коваріація x та y
- ▶ $c_1 = (k_1 L)^2, c_2 = (k_2 L)^2$
- ▶ L - динамічний діапазон пікселів
- ▶ $k_1 = 0.01$ та $k_2 = 0.03$ - константи.

Шар мережі (активаційна функція)	Розмірність
Енкодер	
Dense (Relu)	784×64
Dense (Relu)	64×32
Декодер	
Dense (Sigmoid)	32×784

Табл.: Архітектура щільної нейронної мережі для автоенкодера.

Автоенкодер

Автоенкодер



Автоенкодер для видалення шуму

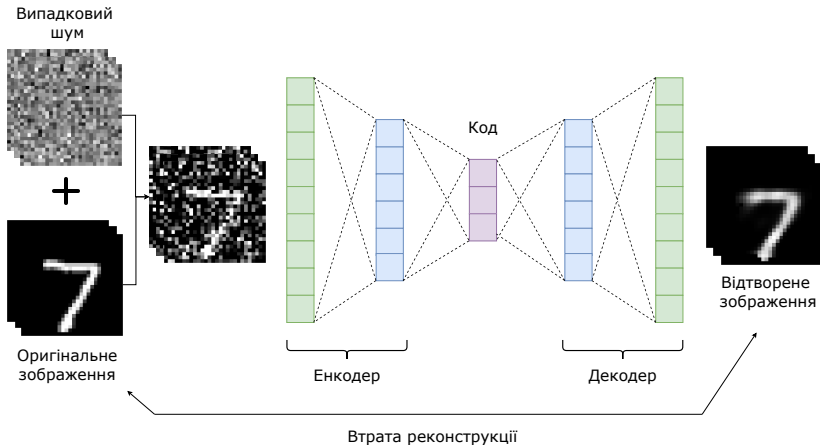


Рис.:

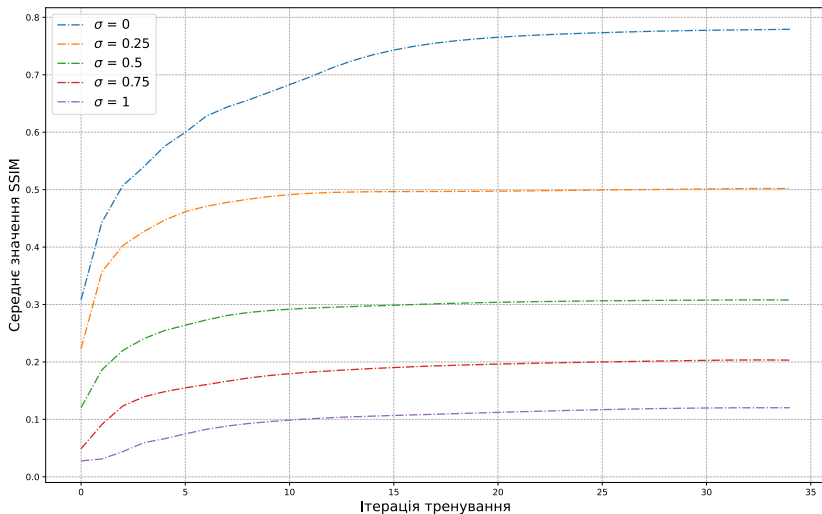


Рис.: Графік залежності усередненої SSIM оцінки для тестового датасету від кількості ітерацій тренування. σ відповідає середньоквадратичному відхиленню гаусівського шуму.

Результати натренованих моделей автоенкодера для видалення шуму можна бачити на зображенні 3.

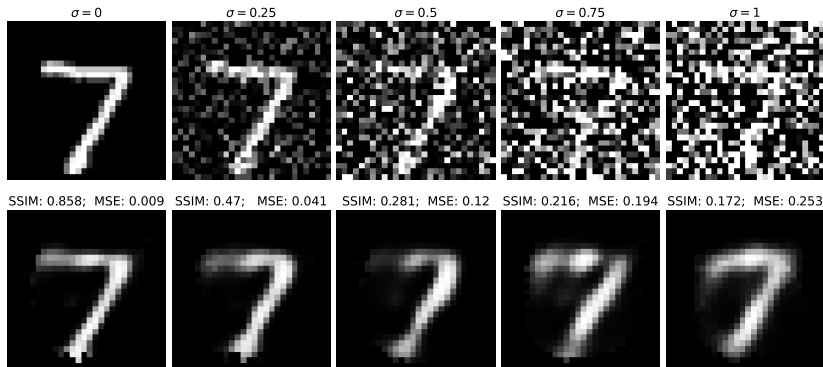


Рис.: Порівняння точності реконструкції зображень автоенкодером для різної величини стандартного відхилення σ білого шуму Гауса.

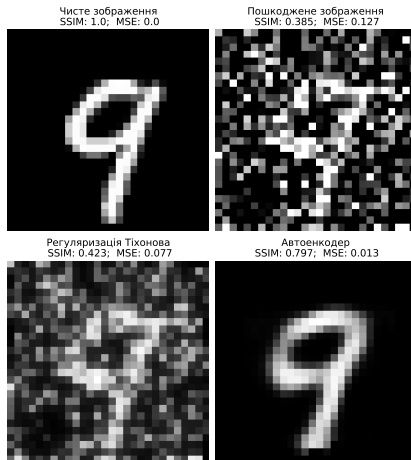






Рис.: Порівняння видалення шуму за допомогою автоенкодера з класичним методом оснований на регуляризації.

-  Gregory Ongie та ін. *Deep Learning Techniques for Inverse Problems in Imaging*. 2020. arXiv: 2005.06001 [eess.IV].
-  Ian Goodfellow, Yoshua Bengio та Aaron Courville. *Deep Learning*. <http://www.deeplearningbook.org>. MIT Press, 2016.
-  Jonas Adler та Ozan Öktem. “Solving ill-posed inverse problems using iterative deep neural networks”. в: *Inverse Problems* 33.12 (листоп. 2017), с. 124007. ISSN: 1361-6420. DOI: 10.1088/1361-6420/aa9581. URL: <http://dx.doi.org/10.1088/1361-6420/aa9581>.
-  Dor Bank, Noam Koenigstein та Raja Giryes. *Autoencoders*. 2021. arXiv: 2003.05991 [cs.LG].