

Використовування прихованих марківських моделей в задачах покращення якості даних

Виконав студент 2 курсу

Магістерської програми «Системний Аналіз»

факультету інформатики

Картавий М. О.

Проблема покращення якості даних

- З кожним днем обсяги інформації у світі зростають
- Використовувати всюди нейронні мережі – задорого
- Мережевий трафік з кожним днем навантажується все більше
- Тенденція зросту можливостей процесорів сповільнюється
- Розвиток нових алгоритмів сповільнився

Постановка задачі

- Створити систему, котра буде покращувати якість даних, на прикладі зображень, з використанням прихованих марківських моделей, а саме:
 - Покращувати якість даних з задовільним результатом
 - Витратити мінімум ресурсів
 - Мати можливість розпаралелювання для кращої швидкодії та масштабування

Прихована марківська модель

статистична марковська модель, у якій система, що моделюється, розглядається як марковський процес із прихованими станами

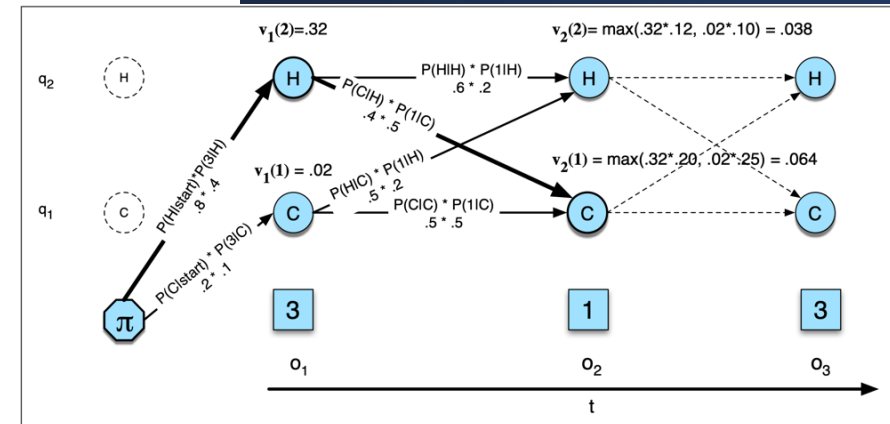
ймовірність певного стану залежить тільки від попереднього стану $P(q_i | q_1 \dots q_{i-1}) = P(q_i | q_{i-1})$

ймовірність вихідного спостереження o_i залежить лише від стану, який викликав спостереження q_i , а не від будь-яких інших станів чи будь-яких інших спостережень

$$P(o_i | q_1 \dots q_i, \dots, q_T, o_1, \dots, o_i, \dots, o_T) = P(o_i | q_i)$$

Декодування прихованих станів алгоритмом Вітербі

- 1) Решітка Вітербі для обчислення найкращого шляху через прихований простір станів для подій-пікселів 3-1-3
- 2) Обчислення в кожній комірці слідує рівнянню $v_t(j) = \max_{1 \leq i \leq N-1} v_{t-1}(i) a_{ij} b_j(o_t)$
- 3) Отримана ймовірність, виражена в кожній комірці, є рівнянням $v_t(j) = P(q_0, q_1, \dots, q_{t-1}, o_1, o_2, \dots, o_t, q_t = j | \lambda)$.

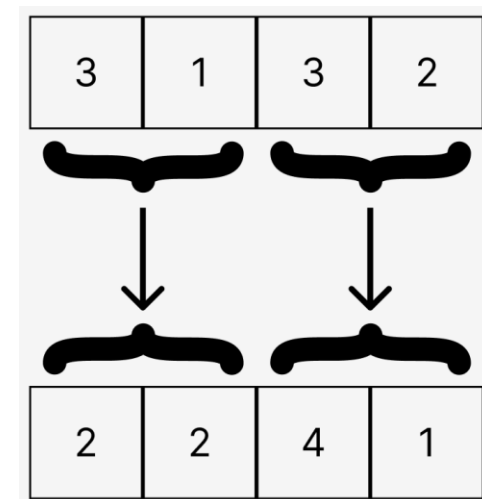
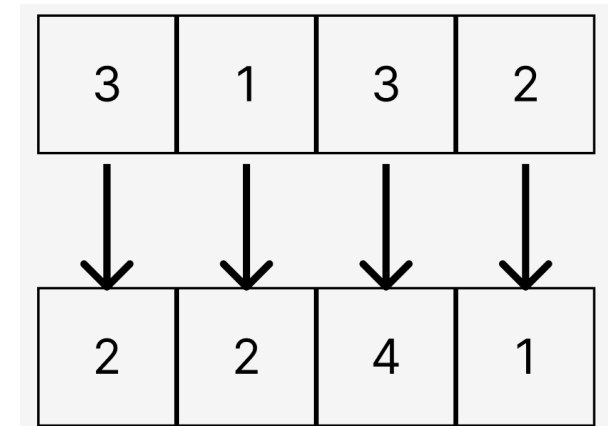


Запропоновані алгоритми з використанням прихованих марківських моделей для задач покращення якості зображень

Номер алгоритму	Опис алгоритму	Проблеми	Стаття для ознайомлення
1	Використовування 2D ПММ з кодуванням стану як набору пікселів розміром 2*2	Використання великої кількості ресурсів	shorturl.at/iGL05
2	Створення бази патчів 3*3, підбору підходящих патчів під кожну зону зображення і вибору найкращою за допомогою ПММ	Потребує створення великої бази даних для покриття всіх можливих кейсів	https://www.researchgate.net/publication/301679307_A_2D_hidden_Markov_model_for_patch-based_super_resolution

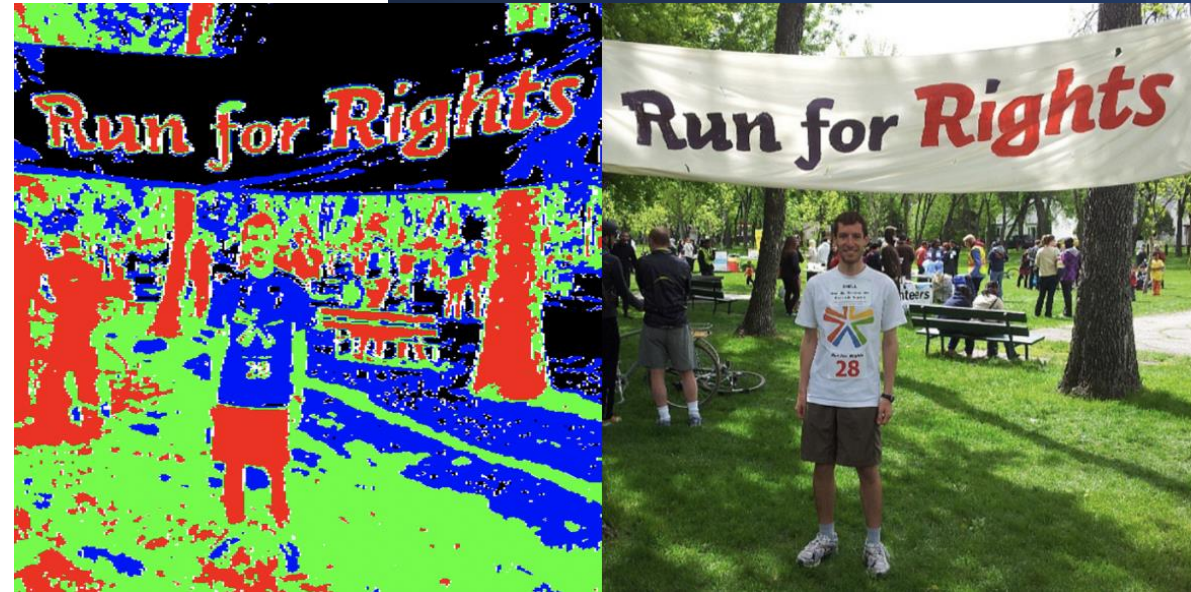
Підходи до вирішення

- Представлення стану як значення конкретного пікселю
- Представлення стану як значення набору пікселів



Розбиття на зони для передбачення

- Оптимальним вибором по кількості моделей та проміжках є співвідношення 4-х моделей ПММ та проміжок з 64 елементів
- Ми маємо 4-и основні зони на зображенні певного каналу(червоний, зелений, синій)
- поділ зон для передбачення
 - значення пікселів 0-64, червоний колір
 - значення пікселів 64-128, зелений колір
 - значення пікселів 128-192, синій колір
 - значення пікселів 192-256, чорний колір
 - білі зони – не належать жодному з проміжків



Побудова ПММ

HR датасет - використовувалися 4600 зображень з роздільною здатністю $300*300$, з різними типами об'єктів та різними показниками різкості, контрастності, яркості, тощо

LR датасет - використовувалися ці ж самі 4600 зображень зменшених до роздільної здатності $100*100$ та за допомогою бікубічної інтерполяції збільшено роздільну здатність до $300*300$

валідаційний датасет HR - використовувалися 1000 зображень з роздільною здатністю $300*300$, з різними типами об'єктів та різними показниками різкості, контрастності, яркості, тощо

валідаційний датасет LR - зображень використовувалися ці ж самі 1000 зображень зменшених до роздільної здатності $100*100$ та за допомогою бікубічної інтерполяції збільшено роздільну здатність до $300*300$

Побудова ПММ

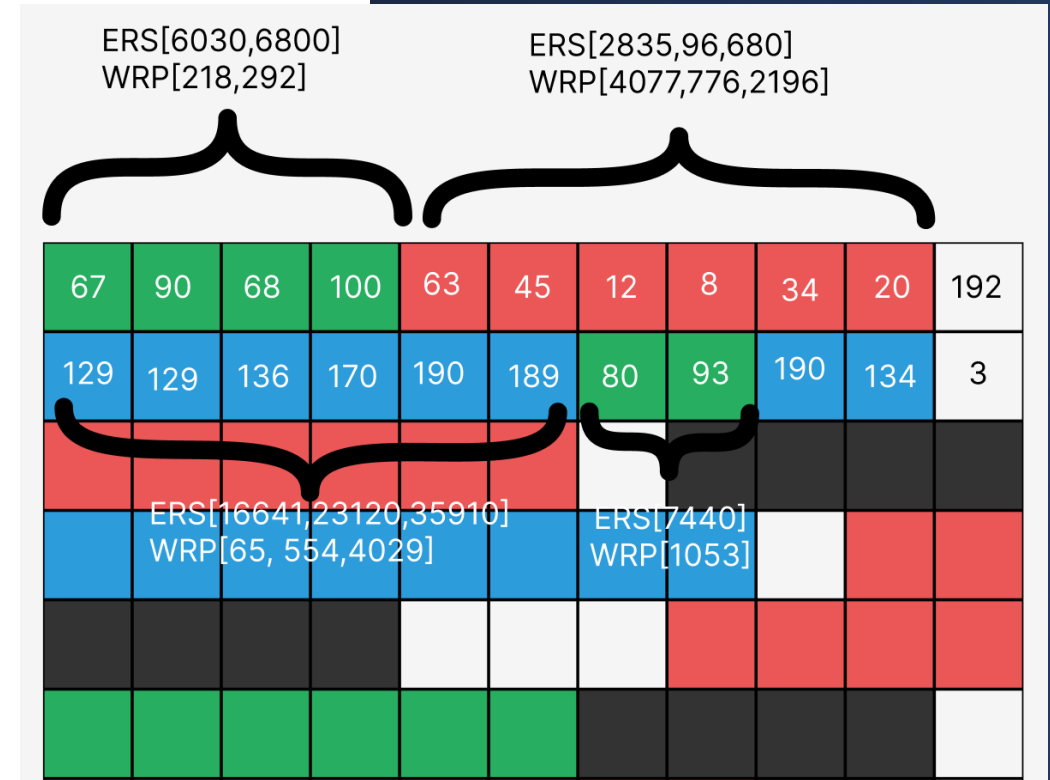
матриця переходів $A = \{M \times N\}$, $x_{mn} \in 0, 1, 2, \dots, 4095$ |
 x_{mn} – стан, що кодується двома сусідніми пікселями
по вісі X

Для пікселів, значення котрих завжди в межах від K
до K+64, де K – це стартова точка для рахунку для
певної зони передбачення, кількість станів як
прихованих так і спостережуваних буде завжди
дорівнювати 4096

стани будуть кодуватися різними проміжками: 0-64,
64-128, 128-192, 192-255

Алгоритм відновлення зображення

- Фіксуємо матрицю кожного каналу та кодуємо пікселі по вісі X по два сусідні пікселі в набір кодованих станів. Кожен кодований стан буде на проміжку від 0 до 65536, позначимо цей проміжок як ERS – encoded range state
- за кожним кодованим станом закріплено його робочий проміжок для передбачення, позначимо як WRP – working range for prediction, на котрому працює модель, це проміжок від 0 до 4096

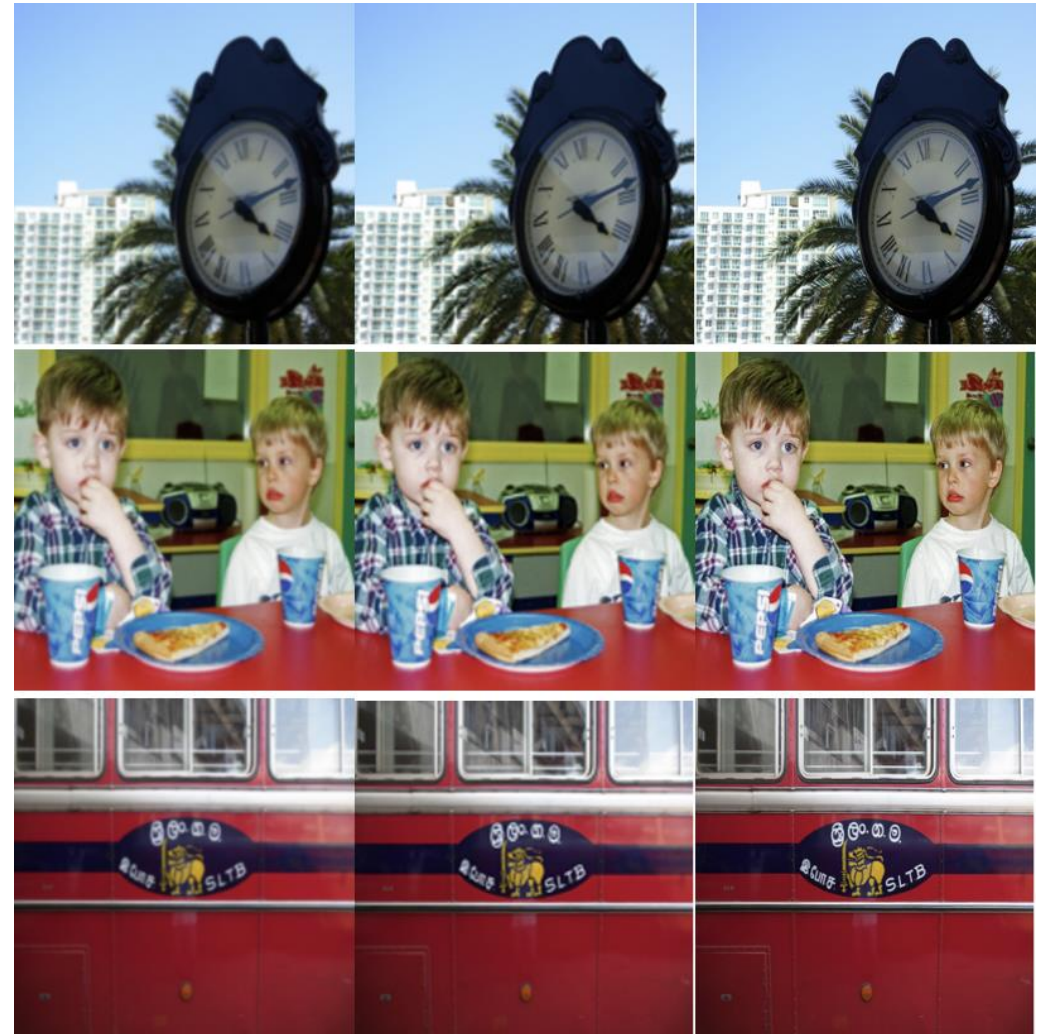


Алгоритм відновлення зображення

- при проходженні в циклі по 2 пікселі, ми кодуємо їх на проміжку від 0 до 65536 і зберігаємо в матрицю, така матриця для зображення розміром $M \times N$, де M – ширина зображення, N – висота, буде мати розмір $M/2 \times N$
- при проходженні в циклі цією матрицею кожного разу ми шукаємо найдовшу послідовність одного з проміжків ERS, після цього ми її (послідовність) переводимо в WRP і передбачаємо певною моделлю, що закріплена за кожним з проміжків
- передбачення з моделі складаємо переводимо в набір пікселів від 0 до 255 та складаємо в матрицю для відновлення
- виконуємо передбачення для кожного з каналів: червоний, зелений, синій та будуємо загальне зображення роздільної здатності, що і зображення високої якості
- порівнюємо MSE для зображення високої якості та зображення, що було розтягнене за допомогою бікубічної інтерполяції та MSE для зображення високої якості та зображення, що було відновлено нашим алгоритмом та порівнюємо результати

Отримані результати

Номер зображення	MSE для бікубічної інтерполяції	MSE для нашого алгоритму	Відсоток покращення
1	166.9	149.2	11.4
2	129.77	116.3	11.2
3	205.68	184.4	11.3
Усереднення	171.5	152.63	11.003



Способи покращення результатів

- визначити семантику зображення: чи є зображення однотонним, чи багато об'єктів на зображенні, тощо - тренувати модель для певного типу зображення
- Використовувати у зв'язці з нейронними мережами, де перший шар покращення зображення - наш алгоритм, другий – нейронна мережа
- Використовувати інший підхід до кодування станів

Висновки

- Отримали результати, згідно з якими ми маємо в середньому 11% покращення
- це не є задовільним результатом, котрий можна було б використовувати в додатках
- співвідношення результатів цього методу до його продуктивності не є високим
- якщо алгоритм Вітербі покращать – менша складність алгоритму або краще передбачення станів, то дослідження можна буде відновити в цьому напрямку

Дякую за увагу!