Обробка зображень і мультимедіа

Олег Гутік

Лекція 11: Стиснення зображень, V. Перетворення зображень

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \rightarrow 96 \times 12 = 1152 \rightarrow MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \rightarrow 96 \times 12 = 1152 \rightarrow MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Поняття перетворення широко застосовується у математиці. З його допомогою розв'язуються багато завдань у різних галузях науки. Основна ідея полягає у зміні математичної величини (числа, вектора, функції або іншого об'єкта) з метою надання їй іншої форми, в якій вона має, можливо, незвичний вигляд, але має корисні властивості. Перетворена величина використовується при розв'язуванні задачі або при здійсненні деяких обчислень, після чого до результату застосовується обернене перетворення для повернення до початкової форми.

Простим ілюстративним прикладом можуть бути арифметичні операції над числами, записаними в римській системі числення. Стародавні римляни, мабуть, знали, як оперувати з такими числами, однак, якщо потрібно, скажімо, перемножити два числа, записані в римській системі числення, то зручніше перетворити їх у сучасну (арабську) форму, перемножити, а потім зобразити результат у вримській системі числення. Ось простий приклад:

$$XCVI \times XII \to 96 \times 12 = 1152 \to MCLII.$$

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

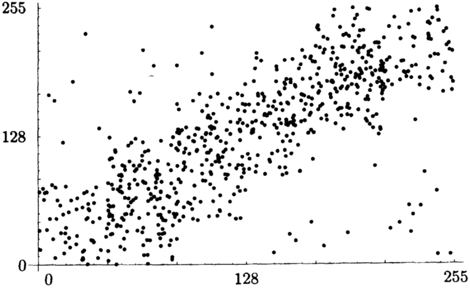
Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.

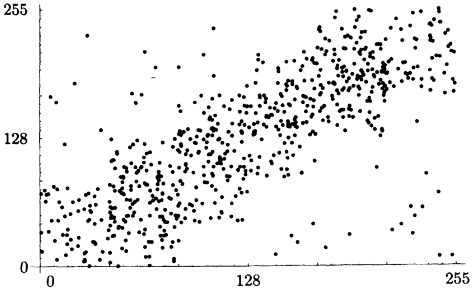
Зображення можна стиснути, перетворюючи його пікселі (які корельовані) на такі, що будуть декорельованими. Відбудеться стиснення, якщо нові величини будуть, в середньому менше початкових. Стиснення з втратою якості можна зробити за допомогою квантування результату перетворення. Декодер читає стислий файл і відновлює (точно чи приблизно) вихідні дані, застосовуючи обернене перетворення. У подальших лекціях розглядаються ортогональні перетворення.

Слово *декорелювання* означає, що перетворені величини є статистично незалежними. В результаті їх можна кодувати незалежно, що дозволяє побудувати простішу статистичну модель. Образ можна стиснути, якщо в ньому є певна надмірність. Надмірність походить від кореляції пікселів. Якщо перевести образ у зображення, в якому пікселі декорелювані, то одночасно відбудеться видалення надмірності й образ буде повністю стиснутий.



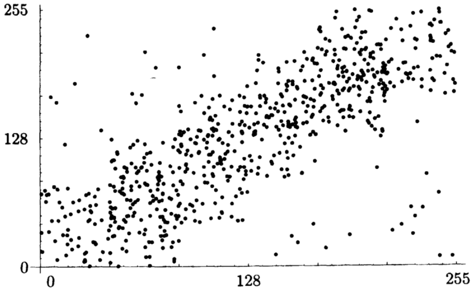
(a)

Почнемо з простого прикладу, в якому образ сканується растровим способом (тобто рядок за рядком) і групується в пари прилеглих пікселів. Оскільки пікселі корельовані, то два пікселі (x, y) у парі, як правило, мають близькі значення. Розглянемо тепер ці пари як точок на площині і відзначимо їх на графіку. Відомо, що точки вигляду (x, x) лежать на прямій з нахилом 45°, рівняння якої має вигляд $y = x$, а тому очікується, що всі точки будуть сконцентровані біля цієї прямої. На рис. (а) зображено такий графік для типового зображення де значення пікселів лежать в інтервалі $[0, 255]$. Більшість точок зображують "хмару" біля цієї лінії, і лише деякі точки лежить далеко від діагоналі.



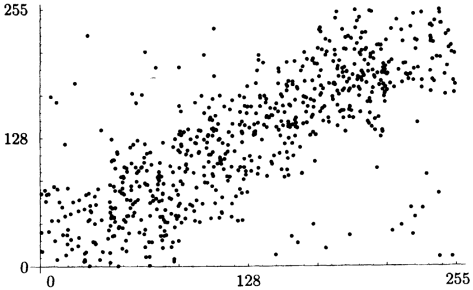
(a)

Почнемо з простого прикладу, в якому образ сканується растровим способом (тобто рядок за рядком) і групується в пари прилеглих пікселів. Оскільки пікселі корельовані, то два пікселі (x, y) у парі, як правило, мають близькі значення. Розглянемо тепер ці пари як точок на площині і відзначимо їх на графіку. Відомо, що точки вигляду (x, x) лежать на прямій з нахилом 45°, рівняння якої має вигляд $y = x$, а тому очікується, що всі точки будуть сконцентровані біля цієї прямої. На рис. (a) зображено такий графік для типового зображення де значення пікселів лежать в інтервалі $[0, 255]$. Більшість точок зображують "хмару" біля цієї лінії, і лише деякі точки лежить далеко від діагоналі.



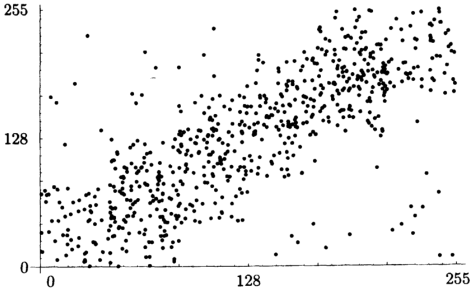
(a)

Почнемо з простого прикладу, в якому образ сканується растровим способом (тобто рядок за рядком) і групується в пари прилеглих пікселів. Оскільки пікселі корельовані, то два пікселі (x, y) у парі, як правило, мають близькі значення. Розглянемо тепер ці пари як точок на площині і відзначимо їх на графіку. Відомо, що точки вигляду (x, x) лежать на прямій з нахилом 45°, рівняння якої має вигляд $y = x$, а тому очікується, що всі точки будуть сконцентровані біля цієї прямої. На рис. (a) зображено такий графік для типового зображення де значення пікселів лежать в інтервалі $[0, 255]$. Більшість точок зображують "хмару" біля цієї лінії, і лише деякі точки лежить далеко від діагоналі.



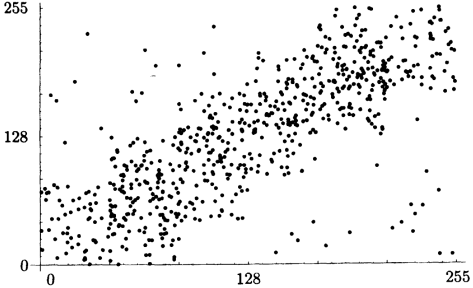
(a)

Почнемо з простого прикладу, в якому образ сканується растровим способом (тобто рядок за рядком) і групується в пари прилеглих пікселів. Оскільки пікселі корельовані, то два пікселі (x, y) у парі, як правило, мають близькі значення. Розглянемо тепер ці пари як точок на площині і відзначимо їх на графіку. Відомо, що точки вигляду (x, x) лежать на прямій з нахилом 45°, рівняння якої має вигляд $y = x$, а тому очікується, що всі точки будуть сконцентровані біля цієї прямої. На рис. (a) зображено такий графік для типового зображення де значення пікселів лежать в інтервалі $[0, 255]$. Більшість точок зображують "хмару" біля цієї лінії, і лише деякі точки лежить далеко від діагоналі.



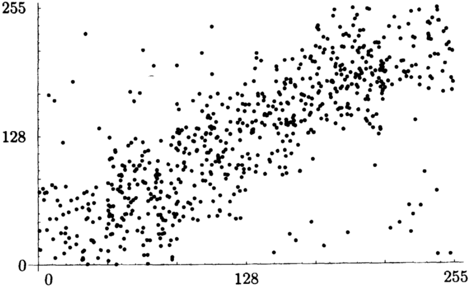
(a)

Почнемо з простого прикладу, в якому образ сканується растровим способом (тобто рядок за рядком) і групується в пари прилеглих пікселів. Оскільки пікселі корельовані, то два пікселі (x, y) у парі, як правило, мають близькі значення. Розглянемо тепер ці пари як точок на площині і відзначимо їх на графіку. Відомо, що точки вигляду (x, x) лежать на прямій з нахилом 45°, рівняння якої має вигляд $y = x$, а тому очікується, що всі точки будуть сконцентровані біля цієї прямої. На рис. (а) зображено такий графік для типового зображення де значення пікселів лежать в інтервалі $[0, 255]$. Більшість точок зображують "хмару" біля цієї лінії, і лише деякі точки лежить далеко від діагоналі.



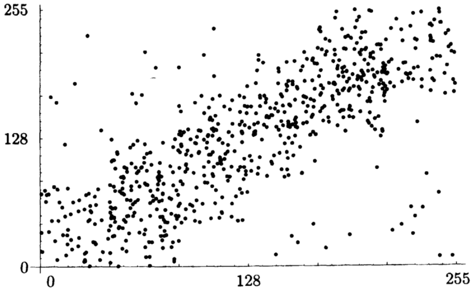
(a)

Почнемо з простого прикладу, в якому образ сканується растровим способом (тобто рядок за рядком) і групується в пари прилеглих пікселів. Оскільки пікселі корельовані, то два пікселі (x, y) у парі, як правило, мають близькі значення. Розглянемо тепер ці пари як точок на площині і відзначимо їх на графіку. Відомо, що точки вигляду (x, x) лежать на прямій з нахилом 45°, рівняння якої має вигляд $y = x$, а тому очікується, що всі точки будуть сконцентровані біля цієї прямої. На рис. (a) зображено такий графік для типового зображення де значення пікселів лежать в інтервалі $[0, 255]$. Більшість точок зображують "хмару" біля цієї лінії, і лише деякі точки лежить далеко від діагоналі.



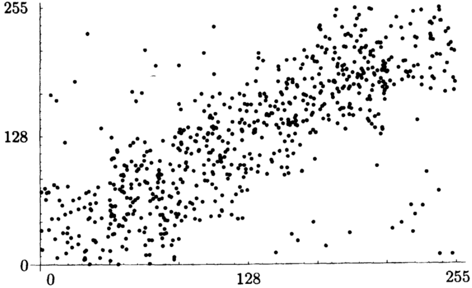
(a)

Почнемо з простого прикладу, в якому образ сканується растровим способом (тобто рядок за рядком) і групується в пари прилеглих пікселів. Оскільки пікселі корельовані, то два пікселі (x, y) у парі, як правило, мають близькі значення. Розглянемо тепер ці пари як точок на площині і відзначимо їх на графіку. Відомо, що точки вигляду (x, x) лежать на прямій з нахилом 45°, рівняння якої має вигляд $y = x$, а тому очікується, що всі точки будуть сконцентровані біля цієї прямої. На рис. (a) зображено такий графік для типового зображення де значення пікселів лежать в інтервалі $[0, 255]$. Більшість точок зображують "хмару" біля цієї лінії, і лише деякі точки лежить далеко від діагоналі.



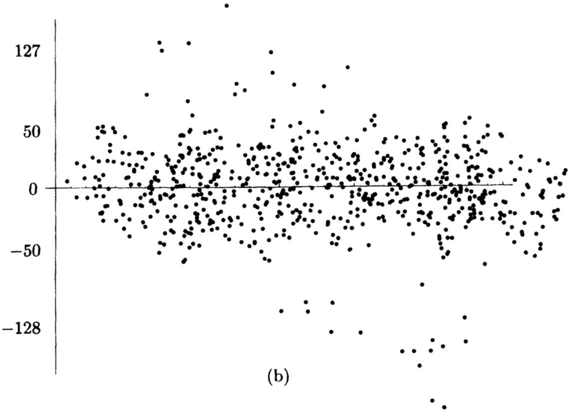
(a)

Почнемо з простого прикладу, в якому образ сканується растровим способом (тобто рядок за рядком) і групується в пари прилеглих пікселів. Оскільки пікселі корельовані, то два пікселі (x, y) у парі, як правило, мають близькі значення. Розглянемо тепер ці пари як точок на площині і відзначимо їх на графіку. Відомо, що точки вигляду (x, x) лежать на прямій з нахилом 45°, рівняння якої має вигляд $y = x$, а тому очікується, що всі точки будуть сконцентровані біля цієї прямої. На рис. (a) зображено такий графік для типового зображення де значення пікселів лежать в інтервалі $[0, 255]$. Більшість точок зображують "хмару" біля цієї лінії, і лише деякі точки лежить далеко від діагоналі.



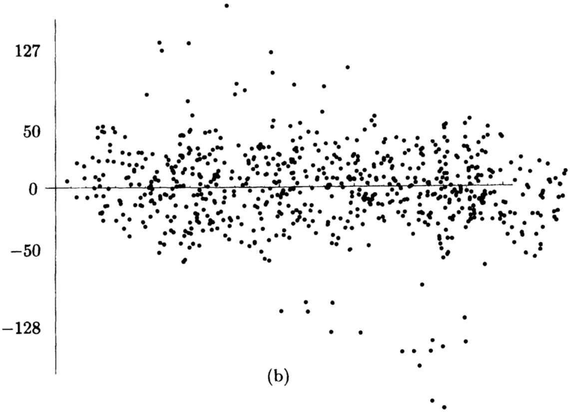
(a)

Почнемо з простого прикладу, в якому образ сканується растровим способом (тобто рядок за рядком) і групується в пари прилеглих пікселів. Оскільки пікселі корельовані, то два пікселі (x, y) у парі, як правило, мають близькі значення. Розглянемо тепер ці пари як точок на площині і відзначимо їх на графіку. Відомо, що точки вигляду (x, x) лежать на прямій з нахилом 45°, рівняння якої має вигляд $y = x$, а тому очікується, що всі точки будуть сконцентровані біля цієї прямої. На рис. (a) зображено такий графік для типового зображення де значення пікселів лежать в інтервалі $[0, 255]$. Більшість точок зображують “хмару” біля цієї лінії, і лише деякі точки лежить далеко від діагоналі.



(b)

Тепер ми перетворимо цю картинку за допомогою повороту на кут 45° за годинниковою стрілкою навколо початку координат так, щоб діагональ лягла на вісь x (рис. (b)).



(b)

Тепер ми перетворимо цю картинку за допомогою повороту на кут 45° за годинниковою стрілкою навколо початку координат так, щоб діагональ лягла на вісь x (рис. (b)).

Це робиться за допомогою простого перетворення

$$\begin{aligned}(x^*, y^*) &= (x, y)\begin{pmatrix} \cos 45^\circ & -\sin 45^\circ \\ \sin 45^\circ & \cos 45^\circ \end{pmatrix} = \\ &= (x, y)\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = \\ &= (x, y)\boldsymbol{R},\end{aligned} \tag{1}$$

де матриця повороту $\boldsymbol{R}$ є ортогональною[1]. Оберненим перетворенням служить поворот на 45° проти годинникової стрілки, який запишеться у вигляді:[2]

$$(x, y) = (x^*, y^*)\boldsymbol{R}^{-1} = (x^*, y^*)\boldsymbol{R}^T = (x^*, y^*)\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}. \tag{2}$$

[1]Квадратна матриця називається *ортогональною*, якщо добуток її однакових рядків (стовпців) дорівнює 1, а різних — 0.

[2]Взяття оберненої матриці до ортогональної робиться, як відомо з курсу лінійної алгебри, за допомогою простої операції транспонування матриць.

Це робиться за допомогою простого перетворення

$$\begin{aligned}(x^*, y^*) &= (x, y)\begin{pmatrix} \cos 45^\circ & -\sin 45^\circ \\ \sin 45^\circ & \cos 45^\circ \end{pmatrix} = \\ &= (x, y)\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = \\ &= (x, y)\boldsymbol{R},\end{aligned} \tag{1}$$

де матриця повороту $\boldsymbol{R}$ є ортогональною[1]. Оберненим перетворенням служить поворот на 45° проти годинникової стрілки, який запишеться у вигляді:[2]

$$(x, y) = (x^*, y^*)\boldsymbol{R}^{-1} = (x^*, y^*)\boldsymbol{R}^T = (x^*, y^*)\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}. \tag{2}$$

[1] Квадратна матриця називається *ортогональною*, якщо добуток її однакових рядків (стовпців) дорівнює 1, а різних — 0.

[2] Взяття оберненої матриці до ортогональної робиться, як відомо з курсу лінійної алгебри, за допомогою простої операції транспонування матриць.

Це робиться за допомогою простого перетворення

$$
\begin{aligned}
(x^*, y^*) &= (x, y)\begin{pmatrix} \cos 45^\circ & -\sin 45^\circ \\ \sin 45^\circ & \cos 45^\circ \end{pmatrix} = \\
&= (x, y)\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = \\
&= (x, y)\boldsymbol{R},
\end{aligned} \tag{1}
$$

де матриця повороту $\boldsymbol{R}$ є ортогональною[1]. Оберненим перетворенням служить поворот на 45° проти годинникової стрілки, який запишеться у вигляді:[2]

$$
(x, y) = (x^*, y^*)\boldsymbol{R}^{-1} = (x^*, y^*)\boldsymbol{R}^T = (x^*, y^*)\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}. \tag{2}
$$

[1] Квадратна матриця називається *ортогональною*, якщо добуток її однакових рядків (стовпців) дорівнює 1, а різних — 0.

[2] Взяття оберненої матриці до ортогональної робиться, як відомо з курсу лінійної алгебри, за допомогою простої операції транспонування матриць.

Це робиться за допомогою простого перетворення

$$
\begin{aligned}
(x^*, y^*) &= (x, y) \begin{pmatrix} \cos 45^\circ & -\sin 45^\circ \\ \sin 45^\circ & \cos 45^\circ \end{pmatrix} = \\
&= (x, y) \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = \\
&= (x, y)\boldsymbol{R},
\end{aligned} \tag{1}
$$

де матриця повороту $\boldsymbol{R}$ є ортогональною[1]. Оберненим перетворенням служить поворот на 45° проти годинникової стрілки, який запишеться у вигляді:[2]

$$
(x, y) = (x^*, y^*)\boldsymbol{R}^{-1} = (x^*, y^*)\boldsymbol{R}^T = (x^*, y^*)\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}. \tag{2}
$$

[1] Квадратна матриця називається *ортогональною*, якщо добуток її однакових рядків (стовпців) дорівнює 1, а різних — 0.

[2] Взяття оберненої матриці до ортогональної робиться, як відомо з курсу лінійної алгебри, за допомогою простої операції транспонування матриць.

Це робиться за допомогою простого перетворення

$$\begin{aligned}(x^*, y^*) &= (x, y)\begin{pmatrix} \cos 45^\circ & -\sin 45^\circ \\ \sin 45^\circ & \cos 45^\circ \end{pmatrix} = \\ &= (x, y)\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = \\ &= (x, y)\boldsymbol{R},\end{aligned} \tag{1}$$

де матриця повороту $\boldsymbol{R}$ є ортогональною[1]. Оберненим перетворенням служить поворот на 45° проти годинникової стрілки, який запишеться у вигляді:[2]

$$(x, y) = (x^*, y^*)\boldsymbol{R}^{-1} = (x^*, y^*)\boldsymbol{R}^T = (x^*, y^*)\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}. \tag{2}$$

[1] Квадратна матриця називається *ортогональною*, якщо добуток її однакових рядків (стовпців) дорівнює 1, а різних — 0.

[2] Взяття оберненої матриці до ортогональної робиться, як відомо з курсу лінійної алгебри, за допомогою простої операції транспонування матриць.

Це робиться за допомогою простого перетворення

$$(x^*, y^*) = (x, y) \begin{pmatrix} \cos 45^\circ & -\sin 45^\circ \\ \sin 45^\circ & \cos 45^\circ \end{pmatrix} = \\ = (x, y) \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = \\ = (x, y)\boldsymbol{R}, \tag{1}$$

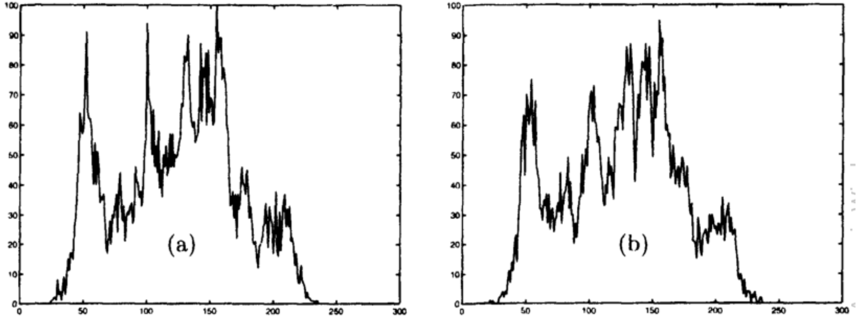
де матриця повороту $\boldsymbol{R}$ є ортогональною[1]. Оберненим перетворенням служить поворот на 45° проти годинникової стрілки, який запишеться у вигляді:[2]

$$(x, y) = (x^*, y^*)\boldsymbol{R}^{-1} = (x^*, y^*)\boldsymbol{R}^T = (x^*, y^*) \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}. \tag{2}$$

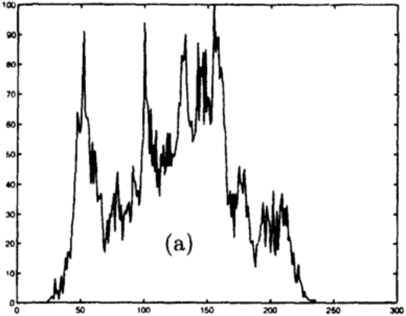
[1] Квадратна матриця називається *ортогональною*, якщо добуток її однакових рядків (стовпців) дорівнює 1, а різних — 0.

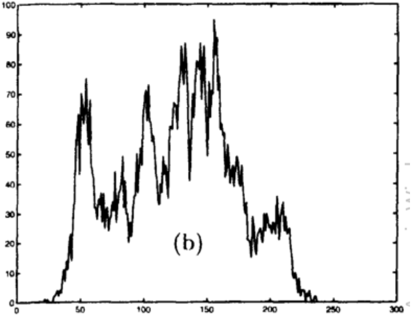
[2] Взяття оберненої матриці до ортогональної робиться, як відомо з курсу лінійної алгебри, за допомогою простої операції транспонування матриць.

Стиснення зображень. Перетворення зображень

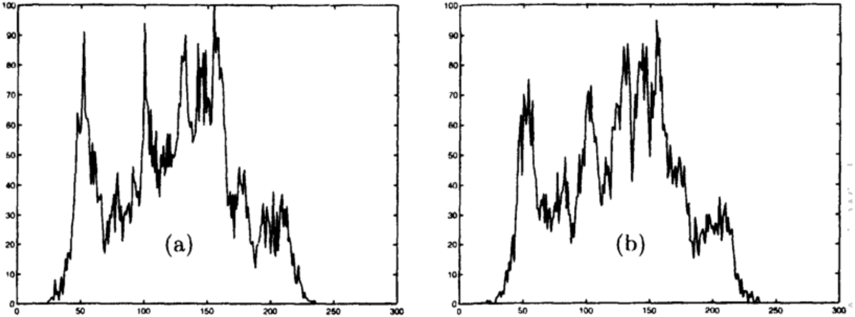


Очевидно, що більшість точок перетвореної “хмари” матимуть координату y, близьку до нуля, а координата x зміниться дуже сильно. Рис. (a),(b) зображують розподіл координат x і y (тобто пікселів з парними та непарними номерами) типового $128 \times 128 \times 8$ напівтонового зображення до обертання. Зрозуміло, що ці розподіли відрізняються не надто.



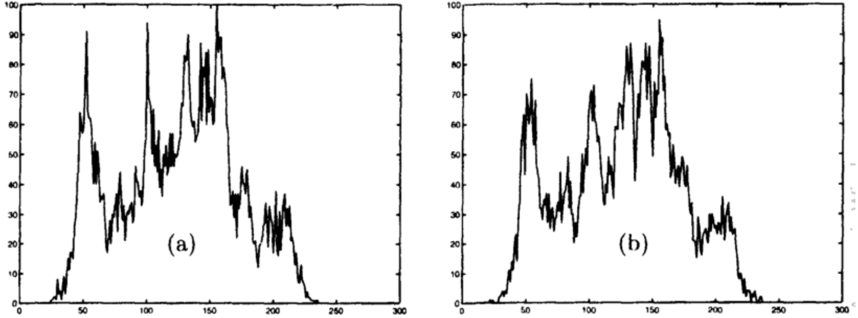


Очевидно, що більшість точок перетвореної "хмари" матимуть координату y, близьку до нуля, а координата x зміниться дуже сильно. Рис. (a),(b) зображують розподіл координат x і y (тобто пікселів з парними та непарними номерами) типового $128 \times 128 \times 8$ напівтонового зображення до обертання. Зрозуміло, що ці розподіли відрізняються не надто.

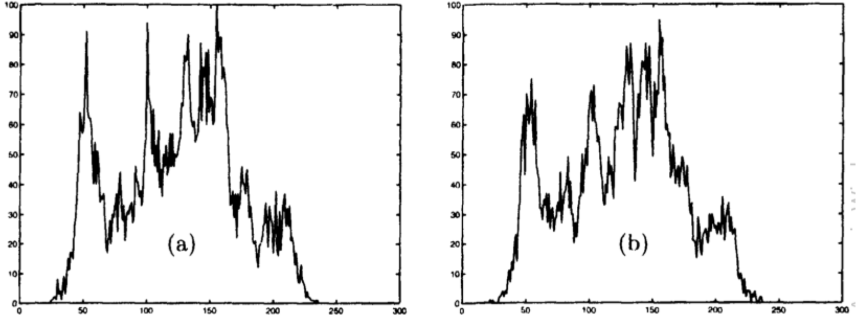


Очевидно, що більшість точок перетвореної "хмари" матимуть координату y, близьку до нуля, а координата x зміниться дуже сильно. Рис. (a),(b) зображують розподіл координат x і y (тобто пікселів з парними та непарними номерами) типового $128 \times 128 \times 8$ напівтонового зображення до обертання. Зрозуміло, що ці розподіли відрізняються не надто.

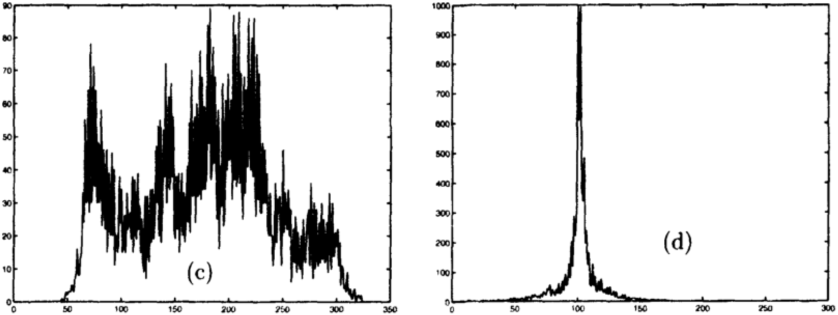
Стиснення зображень. Перетворення зображень



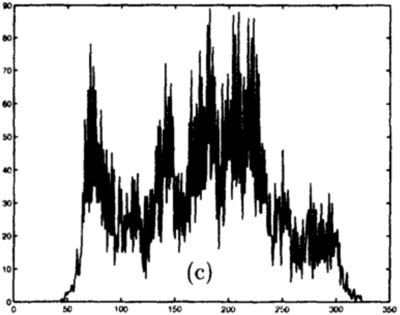
Очевидно, що більшість точок перетвореної “хмари” матимуть координату y, близьку до нуля, а координата x зміниться дуже сильно. Рис. (a),(b) зображують розподіл координат x і y (тобто пікселів з парними та непарними номерами) типового $128 \times 128 \times 8$ напівтонового зображення до обертання. Зрозуміло, що ці розподіли відрізняються не надто.

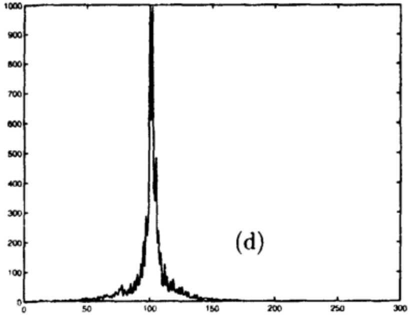


Очевидно, що більшість точок перетвореної "хмари" матимуть координату y, близьку до нуля, а координата x зміниться дуже сильно. Рис. (a),(b) зображують розподіл координат x і y (тобто пікселів з парними та непарними номерами) типового $128 \times 128 \times 8$ напівтонового зображення до обертання. Зрозуміло, що ці розподіли відрізняються не надто.



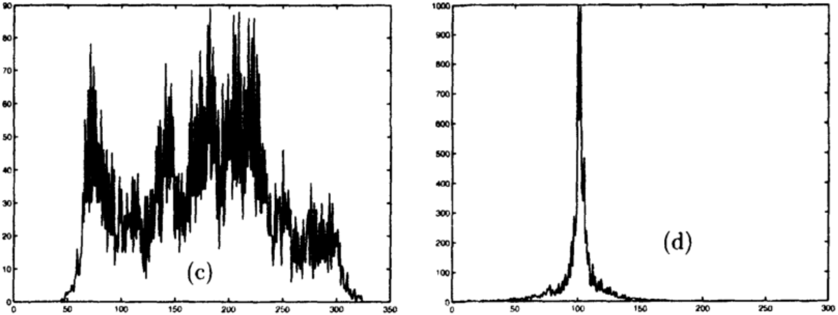
На рис. (c),(d) зображено розподіл координат після повороту. Розподіл координати x майже не змінився (збільшилася дисперсія), тоді як розподіл координати y сконцентрувався біля нуля. На рис. (c) координата y сконцентрована біля 100, але це сталося через зсув графіка вправо, оскільки деякі координати були близькі до -101, і їх довелося зрушити для попадання в масив Matlab, у якого індекс завжди починається з одиниці.



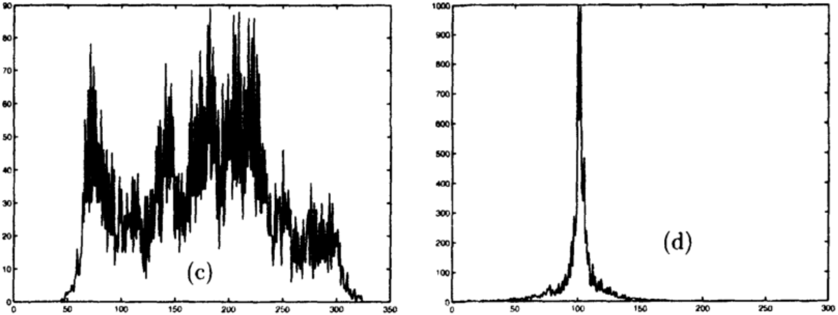


На рис. (c),(d) зображено розподіл координат після повороту. Розподіл координати x майже не змінився (збільшилася дисперсія), тоді як розподіл координати y сконцентрувався біля нуля. На рис. (c) координата y сконцентрована біля 100, але це сталося через зсув графіка вправо, оскільки деякі координати були близькі до -101, і їх довелося зрушити для попадання в масив Matlab, у якого індекс завжди починається з одиниці.

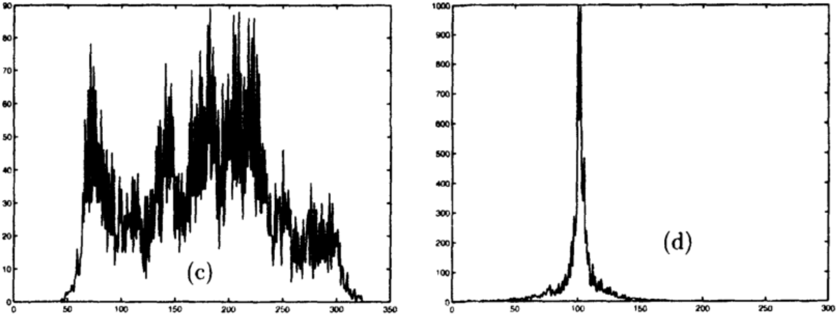
Стиснення зображень. Перетворення зображень



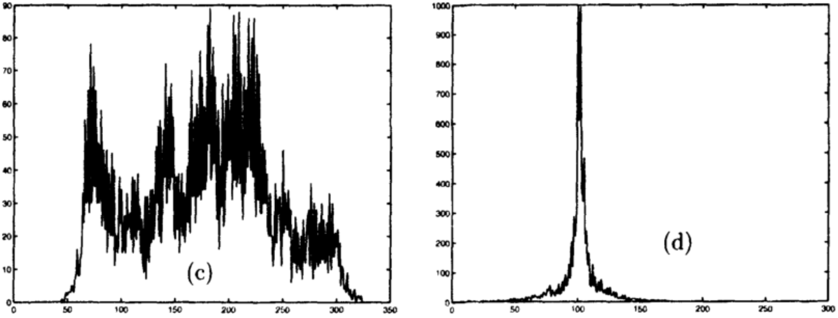
На рис. (c),(d) зображено розподіл координат після повороту. Розподіл координати x майже не змінився (збільшилася дисперсія), тоді як розподіл координати y сконцентрувався біля нуля. На рис. (c) координата y сконцентрована біля 100, але це сталося через зсув графіка вправо, оскільки деякі координати були близькі до -101, і їх довелося зрушити для попадання в масив Matlab, у якого індекс завжди починається з одиниці.



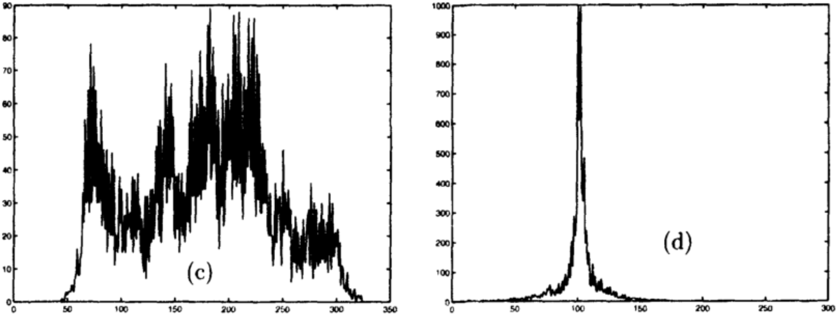
На рис. (c),(d) зображено розподіл координат після повороту. Розподіл координати x майже не змінився (збільшилася дисперсія), тоді як розподіл координати y сконцентрувався біля нуля. На рис. (c) координата y сконцентрована біля 100, але це сталося через зсув графіка вправо, оскільки деякі координати були близькі до -101, і їх довелося зрушити для попадання в масив Matlab, у якого індекс завжди починається з одиниці.



На рис. (c),(d) зображено розподіл координат після повороту. Розподіл координати x майже не змінився (збільшилася дисперсія), тоді як розподіл координати y сконцентрувався біля нуля. На рис. (c) координата y сконцентрована біля 100, але це сталося через зсув графіка вправо, оскільки деякі координати були близькі до -101, і їх довелося зрушити для попадання в масив Matlab, у якого індекс завжди починається з одиниці.



На рис. (c),(d) зображено розподіл координат після повороту. Розподіл координати x майже не змінився (збільшилася дисперсія), тоді як розподіл координати y сконцентрувався біля нуля. На рис. (c) координата y сконцентрована біля 100, але це сталося через зсув графіка вправо, оскільки деякі координати були близькі до -101, і їх довелося зрушити для попадання в масив Matlab, у якого індекс завжди починається з одиниці.



На рис. (c),(d) зображено розподіл координат після повороту. Розподіл координати x майже не змінився (збільшилася дисперсія), тоді як розподіл координати y сконцентрувався біля нуля. На рис. (c) координата y сконцентрована біля 100, але це сталося через зсув графіка вправо, оскільки деякі координати були близькі до -101, і їх довелося зрушити для попадання в масив Matlab, у якого індекс завжди починається з одиниці.

```
filename='lena128'; dim=128;
xdist=zeros(256,1); ydist=zeros(256,1);
fid=fopen(filename,'r');
img=fread(fid,[dim,dim])';
for col=1:2:dim-1
  for row=1:dim
    x=img(row,col)+1; y=img(row,col+1)+1;
    xdist(x)=xdist(x)+1; ydist(y)=ydist(y)+1;
  end
end
figure(1), plot(xdist), colormap(gray) % распред. x&y
figure(2), plot(ydist), colormap(gray) % до поворота
xdist=zeros(325,1); % clear arrays
ydist=zeros(256,1);
for col=1:2:dim-1
  for row=1:dim
    x=round((img(row,col)+img(row,col+1))*0.7071);
    y=round((-img(row,col)+img(row,col+1))*0.7071)+101;
    xdist(x)=xdist(x)+1; ydist(y)=ydist(y)+1;
  end
end
figure(3), plot(xdist), colormap(gray) % распред. x&y
figure(4), plot(ydist), colormap(gray) % после поворота
```

Програма на Matlab, яка будує ці графіки, також наведена на рис.

```
filename='lena128'; dim=128;
xdist=zeros(256,1); ydist=zeros(256,1);
fid=fopen(filename,'r');
img=fread(fid,[dim,dim])';
for col=1:2:dim-1
  for row=1:dim
    x=img(row,col)+1; y=img(row,col+1)+1;
    xdist(x)=xdist(x)+1; ydist(y)=ydist(y)+1;
  end
end
figure(1), plot(xdist), colormap(gray) % распред. x&y
figure(2), plot(ydist), colormap(gray) % до поворота
xdist=zeros(325,1); % clear arrays
ydist=zeros(256,1);
for col=1:2:dim-1
  for row=1:dim
    x=round((img(row,col)+img(row,col+1))*0.7071);
    y=round((-img(row,col)+img(row,col+1))*0.7071)+101;
    xdist(x)=xdist(x)+1; ydist(y)=ydist(y)+1;
  end
end
figure(3), plot(xdist), colormap(gray) % распред. x&y
figure(4), plot(ydist), colormap(gray) % после поворота
```

Програма на Matlab, яка будує ці графіки, також наведена на рис.

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут $45°$ і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Стиснення зображень. Перетворення зображень

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5), (6,7), (12.1, 13.2), (23, 25) \quad \text{та} \quad (32, 29)$$

за годинниковою стрілкою на кут $45°$ і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071, 0), (9.19, 0.7071), (17.9, 0.78), (33.9, 1.41), (43.13, -2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут $45°$ і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут $45°$ і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут $45°$ і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Стиснення зображень. Перетворення зображень

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут 45° і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут 45° і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут 45° і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут $45°$ і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут 45° і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут 45° і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут 45° і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Оскільки координати точок відомі до та після перетворення, то легко обчислити зменшення кореляції. Сума $\sum_i x_i y_i$ називається *перехресною кореляцією точок* (x_i, y_i).

Наступний приклад пояснює зміст цієї величини. Повернемо точки

$$(5,5),(6,7),(12.1,13.2),(23,25) \quad \text{та} \quad (32,29)$$

за годинниковою стрілкою на кут 45° і обчислимо перехресну кореляцію до та після повороту. На рис. наведено програму Matlab, яка обчислює координати повернутих точок.

```
p={{5,5},{6, 7},{12.1,13.2},{23,25},{32,29}};
rot={{0.7071,-0.7071},{0.7071,0.7071}};
Sum[p[[i,1]]p[[i,2]], {i,5}]
q=p.rot
Sum[q[[i,1]]q[[i,2]], {i,5}]
```

Вони дорівнюють

$$(7.071,0),(9.19,0.7071),(17.9,0.78),(33.9,1.41),(43.13,-2.12)$$

(зверніть увагу, що координати y стали малими числами). Видно, що значення перехресної кореляції зменшилося з 1729.72 до повороту до -23.0846 після повороту. Чудове скорочення!

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів.
Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum\limits_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Стиснення зображень. Перетворення зображень

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Тепер можна стиснути образ просто записавши перетворені координати у вихідний файл. Якщо допустима деяка втрата інформації, то можна зробити квантування всіх пікселів, що дасть малі значення пікселів. Можна також записувати в стислий файл усі непарні пікселі (ті, які є x координатами в парах), а за ними записати всі парні пікселі. Ці дві послідовності називаються *векторами коефіцієнтів перетворення*. Друга послідовність складається з малих чисел, і, можливо, після її квантування виникнуть серії нулів, які можна буде ще краще стиснути.

Легко довести, що повна дисперсія пікселів, яка визначається сумою $\sum\limits_i (x_i^2 + y_i^2)$, не змінюється при повороті, оскільки матриця цього перетворення є ортогональною. Проте дисперсія нових координат стала малою, тому зросла дисперсія коефіцієнтів x. Дисперсію іноді називають *енергією розподілу пікселів*. Тому ми можемо сказати, що поворот концентрує енергію в координатах x, за допомогою чого досягається стиснення зображення.

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Стиснення зображень. Перетворення зображень

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4,5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4,5)\boldsymbol{R} = (9,1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9,1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4,5)$. Зробивши те саме для точки $(9,0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4,5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4,5)\boldsymbol{R} = (9,1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9,1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4,5)$. Зробивши те саме для точки $(9,0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2+5^2) - (4.5^2+4.5^2)\right]}{4^2+5^2} = \frac{41-40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4,5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4,5)\boldsymbol{R} = (9,1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2+5^2 = 41 = (9^2+1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9,1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4,5)$. Зробивши те саме для точки $(9,0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{[(4^2+5^2)-(4.5^2+4.5^2)]}{4^2+5^2} = \frac{41-40.5}{41} = 0.012.$$

Стиснення зображень. Перетворення зображень

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{[(4^2 + 5^2) - (4.5^2 + 4.5^2)]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4,5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4,5)\boldsymbol{R} = (9,1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9,1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4,5)$. Зробивши те саме для точки $(9,0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2+5^2)-(4.5^2+4.5^2)\right]}{4^2+5^2} = \frac{41-40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2 + 5^2) - (4.5^2 + 4.5^2)\right]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4, 5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4, 5)\boldsymbol{R} = (9, 1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9, 1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4, 5)$. Зробивши те саме для точки $(9, 0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{[(4^2 + 5^2) - (4.5^2 + 4.5^2)]}{4^2 + 5^2} = \frac{41 - 40.5}{41} = 0.012.$$

Концентрування енергії в одній координаті має й іншу перевагу. Можна робити квантування цієї координати точнішим, ніж квантування другої координати. Такий спосіб квантування призводить до кращого стиску.

Наступний простий приклад ілюструє можливості цього ортогонального перетворення. Почнемо з точки $(4,5)$ з близькими координатами. За допомогою рівняння (1) ця точка переходить у

$$(4,5)\boldsymbol{R} = (9,1)/\sqrt{2} \approx (6.36396, 0.7071).$$

Енергії цих точок дорівнюють:

$$4^2 + 5^2 = 41 = (9^2 + 1^2)/2.$$

Якщо видалити координату (4) вихідної точки, то отримаємо помилку $4^2/41 = 0.39$. Однак, якщо видалити меншу з двох координат перетвореної точки (0.7071), то помилка буде всього-навсього дорівнювати $0.7071^2/41 = 0.012$. Цю помилку можна також обчислити, виходячи з реконструйованої точки. Застосуємо обернене перетворення (рівняння (2)) до точки $(9,1)/\sqrt{2}$. Отримаємо, звісно, вихідну точку $(4,5)$. Зробивши те саме для точки $(9,0)/\sqrt{2}$, отримаємо після заокруглення точку $(4.5, 4.5)$. Різниця енергій вихідної та реконструйованої точок дорівнюватиме тій же малій величині

$$\frac{\left[(4^2+5^2)-(4.5^2+4.5^2)\right]}{4^2+5^2} = \frac{41-40.5}{41} = 0.012.$$

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом $45°$ до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом 45° до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом 45° до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом 45° до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом 45° до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом 45° до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом 45° до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом 45° до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом 45° до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом 45° до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Це просте перетворення легко узагальнити у випадку будь-якого виміру простору. Замість пар можна вибирати трійки точок, триплети. Кожен триплет стає точкою тривимірного простору, а всі точки утворюють "хмару" навколо прямої, яка проходить через початок координат під кутом 45° до кожної координатної осі. Якщо цю пряму повернути так, що вона ляже на вісь x то координати y і z точок "хмари" стануть малими числами. Таке перетворення відбувається за допомогою множення кожної точки на деяку матрицю розміру 3×3 яка, звичайно, є ортогональною. Друга та третя компоненти координат перетворених точок будуть малими числами, а тому координати всіх точок слід розділити на три вектори коефіцієнтів. Для кращого стиснення необхідно, щоб квантування цих векторів коефіцієнтів робилося з різним ступенем точності.

Цей метод легко поширити на більш високі виміри, з тією різницею, що простори, що виходять, вже не можна буде уявити візуально. Однак відповідні матриці перетворень легко виписуються. Єдине, що доводиться враховувати, це те, що вимір не повинен бути надто великим, оскільки результат стиснення на основі повороту залежить від кореляції близьких пікселів. Наприклад, якщо об'єднувати по 24 сусідні пікселі в одну точку 24-вимірного простору, то отримані точки, взагалі кажучи, не лежатимуть у малому околі "прямої під кутом 45° до осей координат", оскільки не буде кореляції між пікселем та його далеким сусідом. Тому після повороту останні 23 координати перетворених точок вже не будуть малими. Спостереження показують, що кореляція пікселів зберігається до виміру вісім, але рідко далі.

Цей метод легко поширити на більш високі виміри, з тією різницею, що простори, що виходять, вже не можна буде уявити візуально. Однак відповідні матриці перетворень легко виписуються. Єдине, що доводиться враховувати, це те, що вимір не повинен бути надто великим, оскільки результат стиснення на основі повороту залежить від кореляції близьких пікселів. Наприклад, якщо об'єднувати по 24 сусідні пікселі в одну точку 24-вимірного простору, то отримані точки, взагалі кажучи, не лежатимуть у малому околі "прямої під кутом 45° до осей координат", оскільки не буде кореляції між пікселем та його далеким сусідом. Тому після повороту останні 23 координати перетворених точок вже не будуть малими. Спостереження показують, що кореляція пікселів зберігається до виміру вісім, але рідко далі.

Цей метод легко поширити на більш високі виміри, з тією різницею, що простори, що виходять, вже не можна буде уявити візуально. Однак відповідні матриці перетворень легко виписуються. Єдине, що доводиться враховувати, це те, що вимір не повинен бути надто великим, оскільки результат стиснення на основі повороту залежить від кореляції близьких пікселів. Наприклад, якщо об'єднувати по 24 сусідні пікселі в одну точку 24-вимірного простору, то отримані точки, взагалі кажучи, не лежатимуть у малому околі "прямої під кутом 45° до осей координат", оскільки не буде кореляції між пікселем та його далеким сусідом. Тому після повороту останні 23 координати перетворених точок вже не будуть малими. Спостереження показують, що кореляція пікселів зберігається до виміру вісім, але рідко далі.

Цей метод легко поширити на більш високі виміри, з тією різницею, що простори, що виходять, вже не можна буде уявити візуально. Однак відповідні матриці перетворень легко виписуються. Єдине, що доводиться враховувати, це те, що вимір не повинен бути надто великим, оскільки результат стиснення на основі повороту залежить від кореляції близьких пікселів. Наприклад, якщо об'єднувати по 24 сусідні пікселі в одну точку 24-вимірного простору, то отримані точки, взагалі кажучи, не лежатимуть у малому околі "прямої під кутом 45° до осей координат", оскільки не буде кореляції між пікселем та його далеким сусідом. Тому після повороту останні 23 координати перетворених точок вже не будуть малими. Спостереження показують, що кореляція пікселів зберігається до виміру вісім, але рідко далі.

Цей метод легко поширити на більш високі виміри, з тією різницею, що простори, що виходять, вже не можна буде уявити візуально. Однак відповідні матриці перетворень легко виписуються. Єдине, що доводиться враховувати, це те, що вимір не повинен бути надто великим, оскільки результат стиснення на основі повороту залежить від кореляції близьких пікселів. Наприклад, якщо об'єднувати по 24 сусідні пікселі в одну точку 24-вимірного простору, то отримані точки, взагалі кажучи, не лежатимуть у малому околі "прямої під кутом 45° до осей координат", оскільки не буде кореляції між пікселем та його далеким сусідом. Тому після повороту останні 23 координати перетворених точок вже не будуть малими. Спостереження показують, що кореляція пікселів зберігається до виміру вісім, але рідко далі.

Цей метод легко поширити на більш високі виміри, з тією різницею, що простори, що виходять, вже не можна буде уявити візуально. Однак відповідні матриці перетворень легко виписуються. Єдине, що доводиться враховувати, це те, що вимір не повинен бути надто великим, оскільки результат стиснення на основі повороту залежить від кореляції близьких пікселів. Наприклад, якщо об'єднувати по 24 сусідні пікселі в одну точку 24-вимірного простору, то отримані точки, взагалі кажучи, не лежатимуть у малому околі "прямої під кутом 45° до осей координат", оскільки не буде кореляції між пікселем та його далеким сусідом. Тому після повороту останні 23 координати перетворених точок вже не будуть малими. Спостереження показують, що кореляція пікселів зберігається до виміру вісім, але рідко далі.

Цей метод легко поширити на більш високі виміри, з тією різницею, що простори, що виходять, вже не можна буде уявити візуально. Однак відповідні матриці перетворень легко виписуються. Єдине, що доводиться враховувати, це те, що вимір не повинен бути надто великим, оскільки результат стиснення на основі повороту залежить від кореляції близьких пікселів. Наприклад, якщо об'єднувати по 24 сусідні пікселі в одну точку 24-вимірного простору, то отримані точки, взагалі кажучи, не лежатимуть у малому околі “прямої під кутом 45° до осей координат”, оскільки не буде кореляції між пікселем та його далеким сусідом. Тому після повороту останні 23 координати перетворених точок вже не будуть малими. Спостереження показують, що кореляція пікселів зберігається до виміру вісім, але рідко далі.

Цей метод легко поширити на більш високі виміри, з тією різницею, що простори, що виходять, вже не можна буде уявити візуально. Однак відповідні матриці перетворень легко виписуються. Єдине, що доводиться враховувати, це те, що вимір не повинен бути надто великим, оскільки результат стиснення на основі повороту залежить від кореляції близьких пікселів. Наприклад, якщо об'єднувати по 24 сусідні пікселі в одну точку 24-вимірного простору, то отримані точки, взагалі кажучи, не лежатимуть у малому околі "прямої під кутом 45° до осей координат", оскільки не буде кореляції між пікселем та його далеким сусідом. Тому після повороту останні 23 координати перетворених точок вже не будуть малими. Спостереження показують, що кореляція пікселів зберігається до виміру вісім, але рідко далі.

Цей метод легко поширити на більш високі виміри, з тією різницею, що простори, що виходять, вже не можна буде уявити візуально. Однак відповідні матриці перетворень легко виписуються. Єдине, що доводиться враховувати, це те, що вимір не повинен бути надто великим, оскільки результат стиснення на основі повороту залежить від кореляції близьких пікселів. Наприклад, якщо об'єднувати по 24 сусідні пікселі в одну точку 24-вимірного простору, то отримані точки, взагалі кажучи, не лежатимуть у малому околі "прямої під кутом 45° до осей координат", оскільки не буде кореляції між пікселем та його далеким сусідом. Тому після повороту останні 23 координати перетворених точок вже не будуть малими.
Спостереження показують, що кореляція пікселів зберігається до виміру вісім, але рідко далі.

Цей метод легко поширити на більш високі виміри, з тією різницею, що простори, що виходять, вже не можна буде уявити візуально. Однак відповідні матриці перетворень легко виписуються. Єдине, що доводиться враховувати, це те, що вимір не повинен бути надто великим, оскільки результат стиснення на основі повороту залежить від кореляції близьких пікселів. Наприклад, якщо об'єднувати по 24 сусідні пікселі в одну точку 24-вимірного простору, то отримані точки, взагалі кажучи, не лежатимуть у малому околі "прямої під кутом 45° до осей координат", оскільки не буде кореляції між пікселем та його далеким сусідом. Тому після повороту останні 23 координати перетворених точок вже не будуть малими. Спостереження показують, що кореляція пікселів зберігається до виміру вісім, але рідко далі.

Дякую за увагу!