НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки

«На правах рукопису» «До захисту допущено»

УДК 004.93.11

В.о. завідувача кафедри
__________ М.В.Грайворонський
“___”_____________2020 р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності: 113 Прикладна математика

на тему: «Сегментація зображення для прискорення алгоритмів стереобачення»

Виконала: студентка 6 курсу, групи ФІ-81мн

Лавягіна Ольга Олексіївна ___________
(підпис)

Науковий керівник к.т.н., доцент Литвинова Т.В. ___________
(підпис)

Консультант ____________ ________________________________ ___________
(назва розділу) (науковий ступінь, вчене звання, , прізвище, ініціали) (підпис)

Рецензент____ ___________
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.
Студент _____________
(підпис)

Київ – 2020 року

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра інформаційної безпеки

Рівень вищої освіти – другий (магістерський) за освітньо-науковою програмою
Спеціальність (освітньо-наукова програма) – 113 Прикладна математика («Математичні методи моделювання, розпізнавання образів та безпеки даних»)

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
__________ М.В.Грайворонський
(підпис)
«____»_____________2020 р.

ЗАВДАННЯ
на магістерську дисертацію студенту

____
(прізвище, ім’я, по батькові)

1. Тема дисертації ____
____
науковий керівник дисертації____,
____,
(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «_____» __________ 2020 р. №

2. Термін подання студентом дисертації _________________

3. Об’єкт дослідження ____
____

4. Предмет дослідження____
____
____
____

5. Перелік завдань, які потрібно розробити ___________________________
____
____
____
____
____

6. Орієнтовний перелік ілюстративного матеріалу _____________________
____
____

7. Орієнтовний перелік публікацій ___________________________________
____
____

8. Консультанти розділів дисертації*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

9. Дата видачі завдання ____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка

Студент ____________ ____________________
(підпис) (ініціали, прізвище)

Науковий керівник дисертації ____________ ____________________
(підпис) (ініціали, прізвище)

* Консультантом не може бути зазначено наукового керівника магістерської дисертації.

РЕФЕРАТ

Диссертация содержит 60 страницы, 32 иллюстраций и список использованной литературы из 13 наименований.

Задача бинокулярного стереозрения является одной из актуальных проблем компьютерного зрения. Она лежит в основе построения карты глубин для получения трёхмерной модели объекта или поверхности.

Цель данной работы — разработка метода ускорения алгоритмов стереозрения с помощью предварительной сегментации изображения.

Для достижения цели были использованы

- сведения из геометрии в компьютерном зрении для правильной постановки задачи;
- сведения из теории оптимизации и алгоритм диффузии для решения оптимизационной задачи;
- алгоритм вычёркивания второго порядка для нахождения наилучшей карты глубин после решения задачи оптимизации.

БИНОКУЛЯРНОЕ СТЕРЕОЗРЕНИЕ, АЛГОРИТМ ДИФФУЗИИ, СЕГМЕНТАЦИЯ, КАРТА ГЛУБИН, ТРЁХМЕРНАЯ МОДЕЛЬ

РЕФЕРАТ

Дисертація містить 60 сторінки, 32 ілюстрацій і 13 джерел літератури.

Задача бінокулярного стереобачення є однією з актуальних проблем комп’ютерного бачення. Вона лежить в основі побудови карти глибин для отримання тривимірної моделі об’єкта чи поверхні.

Метою даної роботи є розробка методу прискорення алгоритмів стереобачення за допомогою сегментації зображення.

Для досягнення мети було використано

- відомості з геометрії в комп’ютерному бачення для правильної постановки задачі;
- відомості з теорії оптимізації та алгоритм дифузії для розв’язання оптимізаційної задачі;
- алгоритм викреслювання другого порядку для знаходження найкращої карти глибин після розв’язання задачі оптимізації.

БІНОКУЛЯРНЕ СТЕРЕОБАЧЕННЯ, АЛГОРИТМ ДИФУЗІЇ, СЕГМЕНТАЦІЯ, КАРТА ГЛИБИН, ТРИВИМІРНА МОДЕЛЬ

ABSTRACT

The thesis contains 60 pages, 32 figures, and 13 references.

Binocular stereo vision problem is one of the topical issues of computer vision. It underlies the construction of a depth map to obtain a three-dimensional model of an object or surface.

This study aims to develop a method for accelerating stereo vision algorithms using preliminary image segmentation.

To perform the study

- information from geometry in computer vision was used to state the problem correctly;
- information from optimization theory and min-sum diffusion algorithm were used to solve an optimization problem;
- relaxation labeling algorithm was used to find the best depth map after solving the optimization problem.

BINOCULAR STEREO VISION, MIN-SUM DIFFUSION ALGORITHM, SEGMENTATION, DEPTH MAP, THREE-DIMENSIONAL MODEL

ЗМІСТ

ВСТУП

Значною мірою робота завдячує Євгенію Валерійовичу Водолазському — старшому науковому співробітнику Відділу обробки та розпізнавання образів Міжнародного науково-навчального центру інформаційних технологій і систем НАН України та МОН України.

Актуальність роботи. За рахунок зростання актуальності автоматичної обробки та аналізу візуальної інформації розробляється багато методів розв'язання задач комп'ютерного бачення, в тому числі стереобачення. Остання задача виникає при проектуванні інтелектуальних робототехнічних комплексів, систем управління рухомими апаратами, при проведенні біомедичних досліджень. Задача стереобачення полягає в побудові карти глибин за двома зображеннями, що лежить в основі отримання тривимірної моделі об'єктів і поверхонь. Алгоритм дифузії — один з популярних методів для розв'язання задачі стереобачення, який часто показує гарні практичні результати, але працює досить повільно. Таким чином, виникає необхідність у розробці більш швидких методів, які дають не гірші результати.

Мета і завдання дослідження.

Об'єкт дослідження — побудова карти глибини за двома зображеннями.

Предмет дослідження — алгоритми стереобачення.

Метою роботи є розробка методу прискорення алгоритмів стереобачення за допомогою сегментації зображення.

Завдання наступні:

1) ознайомитися з задачею стереобачення та існуючими методами її розв'язання;
2) ознайомитися з алгоритмом дифузії, що використовується для розв'язання задачі оптимізації, що виникає при розв'язанні задачі;
3) ознайомитися з алгоритмом викреслювання другого порядку, що використовується для знаходження однієї з найкращих карт глибин після розв'я-

зання оптимізаційної задачі;

4) запропонувати та перевірити метод прискорення алгоритмів розв'язання задачі стереобачення на прикладі алгоритму дифузії;
5) розробити програмну реалізацію алгоритму та його модифікації.

Методи дослідження:

1) збір інформації, опрацювання літератури за темою;
2) теоретична та практична перевірка роботи алгоритму стереобачення;
3) аналіз отриманих результатів.

Наукова новизна одержаних результатів.

В роботі чітко описана постановка задачі стереобачення та її розв'язання за допомогою алгоритму дифузії. Запропоновано новий метод зменшення складності даного алгоритму.

Практичне значення одержаних результатів.

Карти глибин, побудовані за допомогою алгоритмів стереобачення, можна використовувати для відновлення поверхонь за допомогою всього лише двох зображень. Був описаний та перевірений новий спосіб прискорення алгоритмів за допомогою сегментації зображення, за якого не втрачається багато інформації про глибину об'єктів.

Публікації.

XVIII Всеукраїнська науково-практична конференція студентів, аспірантів та молодих вчених «Теоретичні і прикладні проблеми фізики, математики та інформатики».

1 ПОПЕРЕДНІ РОБОТИ ПРИСВЯЧЕНІ ЗАДАЧІ СТЕРЕОБАЧЕННЯ

В першому розділі надано стислий огляд досліджень, що пов'язані із задачею стереобачення. Проводиться аналіз та порівняння деяких існуючих методів розв'язання задачі. Розбір попередніх робіт дає змогу виявити недоліки існуючих рішень та чітко поставити задачу, що розв'язується в другому та третьому розділах дисертації.

1.1 Динамічне програмування

Вважається, що камера не вносить радіальних спотворень, а два зображення отримуються шляхом її ідеального зсуву по горизонталі, тобто пікселям рядка y лівого зображення $L_y : \{1, \ldots, w\} \to C$, де w — ширина зображення, відповідають пікселі рядка y правого зображення $R_y : \{1, \ldots, w\} \to C$, $y = \overline{1, h}$, h — висота зображення, а $C = \{0, \ldots, 255\}$ — множина, що задає інтенсивності пікселів. Вважається, що в кожен рядок лівого зображення проектується свій об'єкт, тому вектори зсувів для всіх рядків можна знаходити окремо шляхом мінімізації штрафної функції

$$E(\boldsymbol{d}, y) = \sum_{x=1}^{w} f_{(x,y)}(d(x)) + \sum_{x=1}^{w} g(d(x), d(x+1)), \tag{1.1}$$

яка називається енергією, де $\boldsymbol{d}$ — вектор зсувів для рядка з номером y, кожен елемент якого може приймати значення з множини $\{0, 1, \ldots, M\}$, $L_y(x)$ — інтенсивність пікселя з координатами (x, y) на лівому зображенні, а $R_y(x)$ — інтенсивність пікселя з координатами (x, y) на правому зображенні. Штрафна функція включає штраф за негладкість карти зсувів g (бінарні штрафи) і невідповідність кольорів зіставлених пікселів на двох зображеннях f (унарні штрафи). Унарні штрафи $f_{(x,y)}(d) = f(L_y(x), R_y(x - d))$ залежать від значень інтенсивностей відповідних пікселів на лівому та правому зображеннях, бінарні штрафи g — від зсувів двох сусідніх по горизонталі пікселів.

Для кожного рядка зображення будується граф (рис. 1.1), де кожен піксель x рядка y є об'єктом, а вершини в об'єктах відповідають можливим зсувам $d\left(x\right)$. Назвемо їх мітками. Сусідніми вважаються лише ті об'єкти, які відповідають сусіднім пікселям (у кожного пікселя є лівий і правий сусідні пікселі). Вага мітки в об'єкті x, що відповідає зсуву $d\left(x\right)$, задається значенням функції $f_{(x,y)}\left(d\right)$, а вага дужки між двома сусідніми об'єктами x і $x+1$ — значенням функції $g\left(d\left(x\right), d\left(x+1\right)\right)$. Методом динамічного програмування знаходиться такий вектор зсувів $\boldsymbol{d}$ для кожного рядка y одного з зображень стереопари, що відповідає найкоротшому шляху на побудованому графі [1].

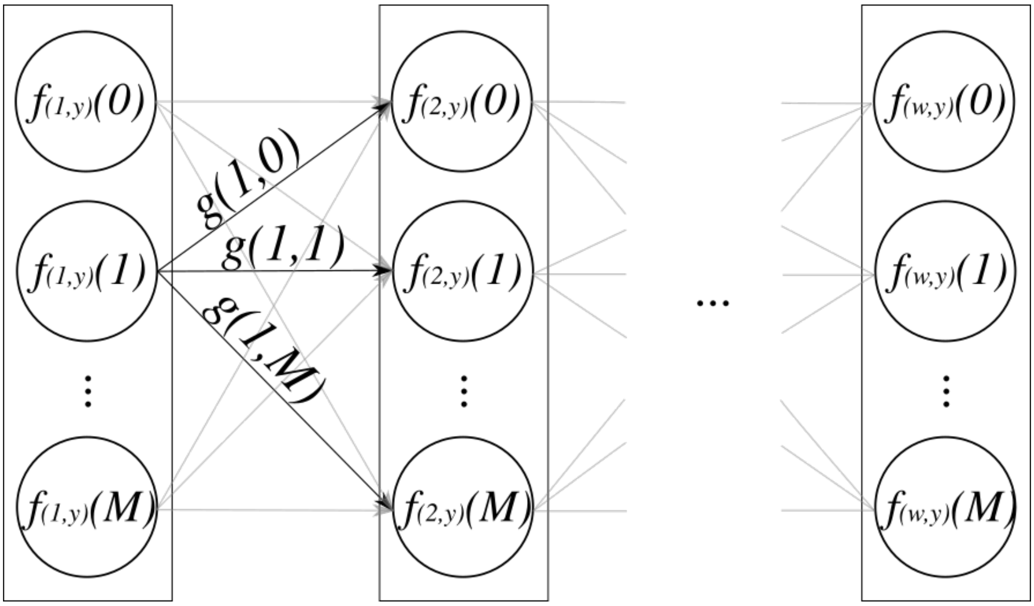
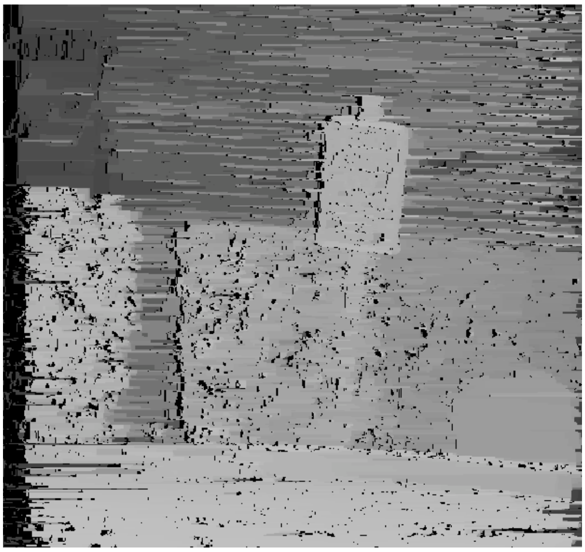


Рисунок 1.1 — Граф, на якому розв'язується задача пошуку мінімального шляху методом динамічного програмування для пошуку вектора зсувів $\boldsymbol{d}$ для рядка y одного з зображень стереопари. Прямокутниками позначені об'єкти, окружностями — вершини, всередині яких зазначені штрафи, на стрілках зазначені штрафи, що накладаються на дужки між вершинами сусідніх об'єктів

Недоліком методу динамічного програмування є те, що він обробляє всі рядки зображення незалежно, тобто не враховує узгодженості між рядками, тому в простому випадку метод динамічного програмування не використовує двовимірність задачі та не дає гладкої карти глибин по вертикалі (рис. 1.2).

(а) Ліве зображення стереопари

(б) Праве зображення стереопари



(в) Карта глибин

Рисунок 1.2 — Результати, отримані методом динамічного програмування (зображення взяті зі статті [1])

Водночас, алгоритм працює досить швидко, а результати можна значно покращити (рис. 1.3), наприклад розглядаючи вісім сусідів пікселя замість двох та використовуючи сегментацію зображення за кольором [2].

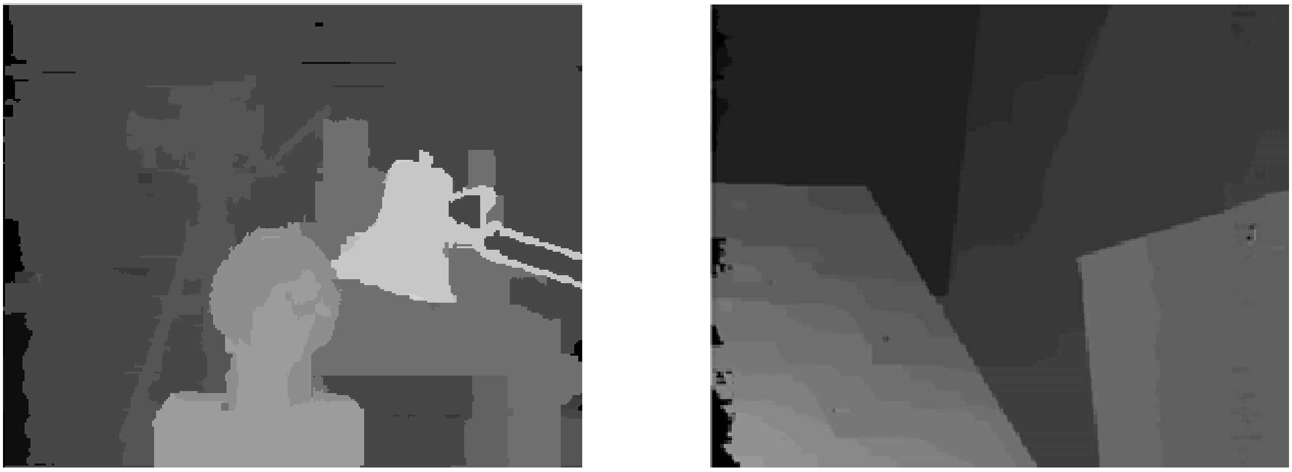
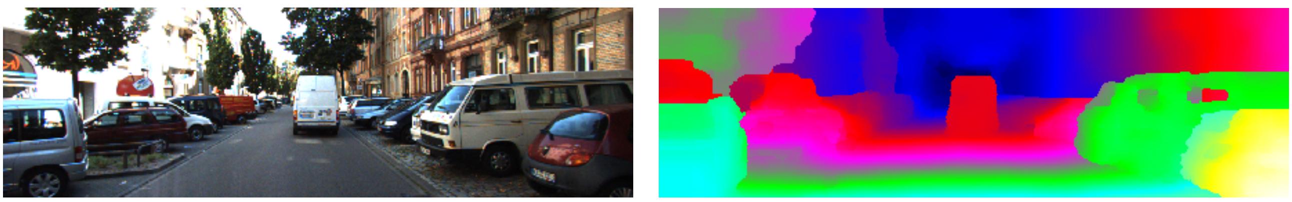


Рисунок 1.3 — Результати, отримані за допомогою покращеного методу динамічного програмування (зображення взяті зі статті [2])

1.2 Нейронні мережі

Нейронна мережа навчається на вхідних даних: парах зображень та відповідній ним карті глибин. Добре розмічені вхідні дані для навчання наявні для зображень дороги. Нейронна мережа, навчена на таких даних може давати погані результати для зображень іншого середовища. На рисунку 1.4 наведено приклад вхідних даних і результуючої карти глибин, що була навчена за допомогою даних з набору «KITTI Stereo» [3].



(а) Ліве зображення

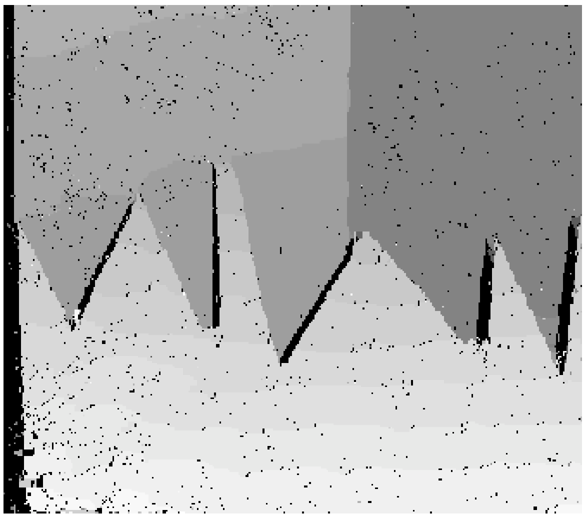
(б) Карта глибин

Рисунок 1.4 — Результати, отримані за допомогою нейронної мережі (зображення взяті зі статті [4])

1.3 Знаходження мінімального розрізу графа

Задачею є мінімізація енергії 1.1. Будується граф-решітка, у якому вершинами є всі пікселі зображення. Назвемо ці вершини основними. Ваги вершин і дуг задаються як в методі динамічного програмування. До графа додається дві допоміжні вершини: джерело та стік, які поєднуються з кожними іншими вершинами графу. Кожній основній вершині має бути поставлена у відповідність мітка — значення зсуву координати пікселя по відношенню до цієї ж координати на іншому зображенні зі стереопари. Застосовується алгоритм α-експансії: спочатку фіксується якась мітка в кожній основній вершині. Також випадковим чином обирається ще одна мітка (однакова для всіх вершин). Шляхом знаходження мінімального розрізу графу, в якому дужки направлені та йдуть від джерела до стоку, в кожній вершині обирається одна з двох міток. Обрані мітки подаються на вхід алгоритму на наступній ітерації. Результати, отримані за допомогою такого методу, представлені на рисунку 1.5.

(а) Ліве зображення



(б) Карта глибин

Рисунок 1.5 — Результати, отримані за допомогою знаходження мінімального розрізу графу (зображення взяті зі статті [5])

Для більш швидкої глобальної оптимізації виконують сегментацію одного із зображень стереопари. На пікселі в одному сегменті накладаються обмеження на зсуви так, що при глобальній оптимізації оновлюється декілька змінних одразу [6].

1.4 Алгоритм дифузії

Будується граф-решітка та мінімізується енергія 1.1 шляхом розв'язання двоїстої задачі [7]. Докладніше алгоритм описано в наступному розділі.

1.5 Постановка задачі

Виходячи з аналізу статей, запишемо постановку задачі, яка буде розв'язана в наступних розділах.

Задане ліве півтонове зображення $L : T \to C$ та праве півтонове зображення $R : T \to C$, де $T = \{(x, y) \mid 1 \leq x \leq w, q \leq y \leq h\}$ — множина координат пікселів зображення, w — ширина зображення (кількість пікселів на зображенні по горизонталі), h — висота зображення (кількість пікселів на зображенні по вертикалі), $C = \{0, \dots, 255\}$ — множина інтенсивностей пікселів. Таким чином, $L(x, y)$ — це інтенсивність пікселя лівого зображення з координатою x по горизонталі та координатою y по вертикалі, а $R(x, y)$ — це інтенсивність пікселя правого зображення з координатою x по горизонталі та координатою y по вертикалі.

Вважається, що L і R — пара зображень нерухомої сцени, що зняті однією й тією ж камерою. При цьому, праве зображення R було отримано при тому ж напрямку зйомки, що й ліве зображення L, але при зсуві камери строго горизонтально праворуч. Це означає, що проекції об'єктів на зображення також зміщуються виключно горизонтально. Будуємо таку модель, згідно якої об'єкти не перекриваються один одним. В експериментах, що наведені в четвертому розділі дисертації, показано, що це не сильно обмежує застосування алгоритму — на практиці його можна використовувати й для об'єктів, які перекриваються.

Таким чином, рядок з вертикальною координатою y на лівому зображенні L відповідає рядку з вертикальною координатою y на правому зображенні R (рис. 1.6). За рахунок паралаксу горизонтальна координата x кожного пікселя лівого зображення $(x, y) \in T$ зсувається вліво на величину $d(x, y)$ для отримання координати відповідного пікселя на правому зображенні $(x - d(x, y), y)$. Множина всіх можливих зсувів $d(x, y)$ горизонтальної координати x пікселя (x, y) дорівнює $D = \{0, \ldots, M\}$, де M — фіксований максимальний зсув.

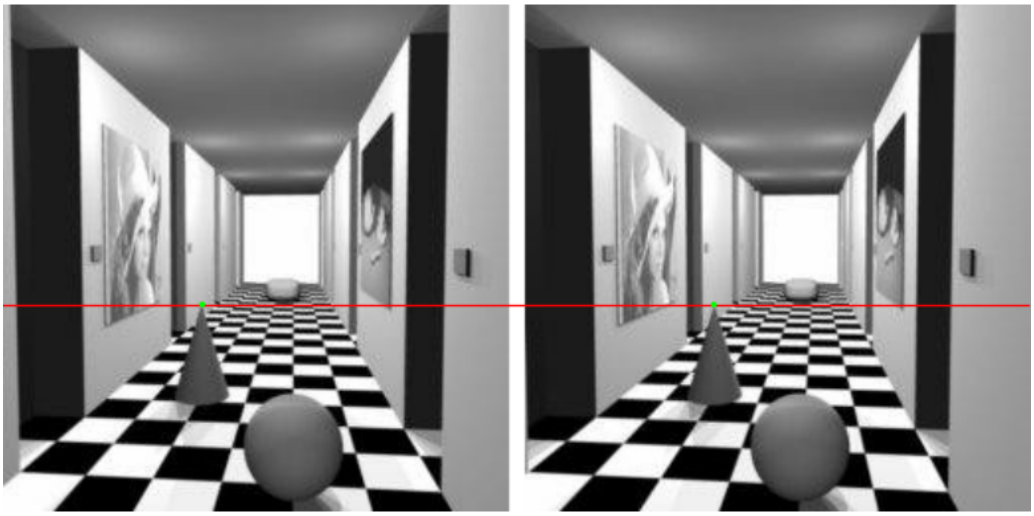


Рисунок 1.6 — Рядок з вертикальною координатою y на лівому зображенні L відповідає рядку з вертикальною координатою y на правому зображенні L (позначено червоною лінією). Зеленими точками позначена вершина конусу на двох зображеннях стереопари, яка лежить на одній горизонтальній лінії

Розглянемо задачу стереобачення в двовимірному світі, де точки простору описуються двома невідомими координатами (x, z) (рис. 1.7). Задача полягає в знаходженні координати z, яка визначає висоту точок в просторі, для всіх пікселів на одному з зображень, знятих камерою. C_1 — точка фокусу камери в першому положенні. Вважаємо, що в ній знаходиться початок координат. C_2 — точка фокусу камери після зсуву її по горизонталі на відстань

b. Двома горизонтальними лініями позначені плівки, або площини, на яких формуються зображення. f — фокусна відстань камери — відстань від точки фокуса камери до плівки. x_1 та x_2 позначають відстані від центрів зображень до променів, що йдуть від точок фокуса камери в двох положеннях до точки в просторі.

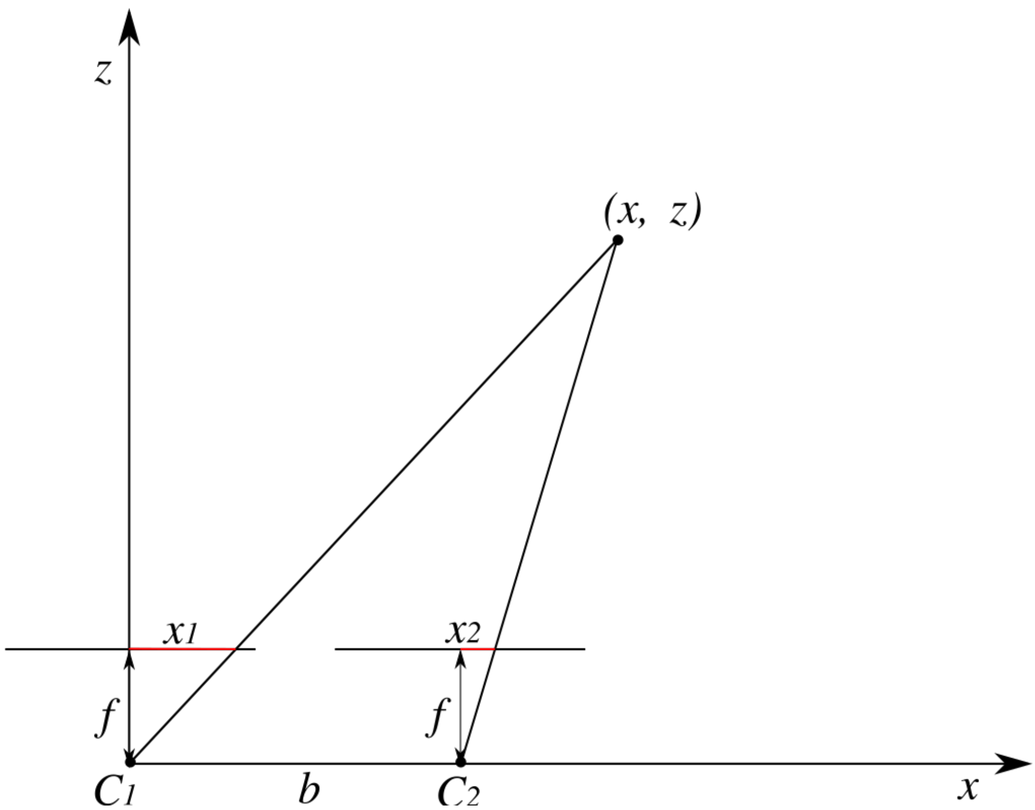


Рисунок 1.7 — Одна точка простору, сфотографована камерою з двох положень

З подібності трикутників $\triangle BXC_1$ та $\triangle C_1F_1P_1$ (рис. 1.8) має місце співвідношення

$$\frac{x}{z} = \frac{x_1}{f}. \tag{1.2}$$

З подібності трикутників $\triangle BXC_2$ та $\triangle C_2F_2P_2$ (рис. 1.8) має місце співвідношення

$$\frac{x-b}{z} = \frac{x_2}{f}. \tag{1.3}$$

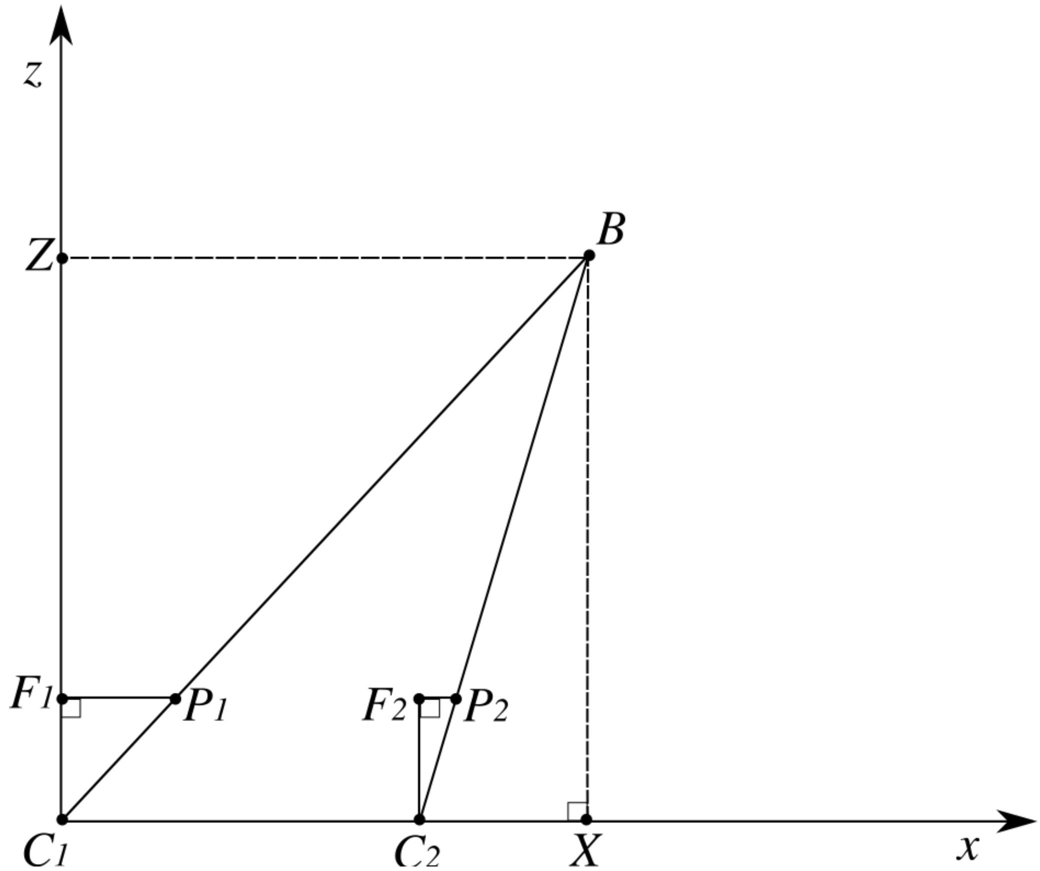


Рисунок 1.8 — Подібні трикутники

Віднімемо співвідношення (1.3) від співвідношення (1.5)

$$\frac{b}{z} = \frac{x_1 - x_2}{f},$$

звідки шукане значення координати z дорівнює

$$z = \frac{b \cdot f}{x_1 - x_2} = \frac{b \cdot f}{d}.$$

Таким чином, щоб знайти положення точки в просторі, необхідно знайти значення паралаксу d, а також знати фокусну відстань камери f та відстань, на яку камера була зрушена по горизонталі b.

Далі розглядається задача пошуку значення паралаксу для всіх пікселів лівого зображення L, тобто задача пошуку карти глибин.

Будується $|T|$-дольний граф-решітка, де кожній долі відповідає один піксель (рис. 1.9). Будемо називати долі графу *об'єктами*.

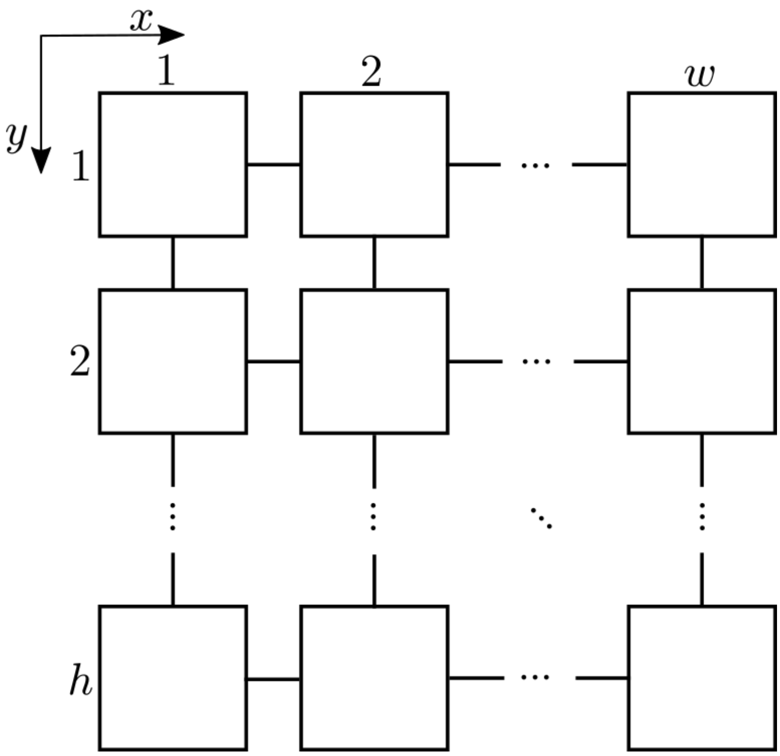


Рисунок 1.9 — $|T|$-дольний граф-решітка. Об'єкти (долі) позначені квадратами. Об'єкти, що поєднані лініями, є сусідніми

Кожен об'єкт $(y, x) \in T$ має $|D|$ вершин (x, y, d), які відповідають всім можливим зсувам $d \in D$ пікселя (x, y) (рис. 1.10). Значення зсувів $d \in D$ будемо називати *мітками*.

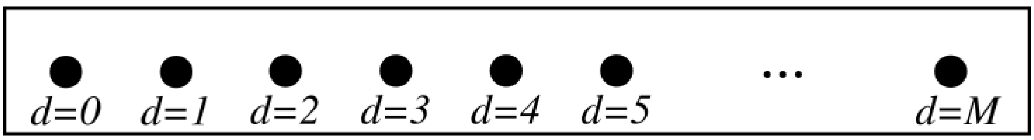


Рисунок 1.10 — Об'єкт (позначений прямокутником) має $|D|$ вершин (позначені кружечками)

На кожну вершину (x, y, d), $d \in D$ в кожній долі графу $(x, y) \in T$ накладається штраф за відхилення між інтенсивностями відповідних пікселів

$$f_{(x,y)}(d) = f\left(L(x, y), R(x - d, y)\right), \tag{1.4}$$

де $f : T \times D \to \mathbb{R}$.

Кожен об'єкт графу має не більше чотирьох сусідніх об'єктів: верхній, правий, нижній і лівий. Об'єкти, що відповідають пікселям на границях зображення, мають по три сусіда, а об'єкти, що відповідають кутовим пікселям, — по два. Нехай $\mathcal{N}(x, y)$ — множина всіх сусідніх об'єктів для об'єкта (x, y). Всі вершини кожного об'єкту поєднані дужками з усіма вершинами в сусідніх об'єктах (рис. 1.11).

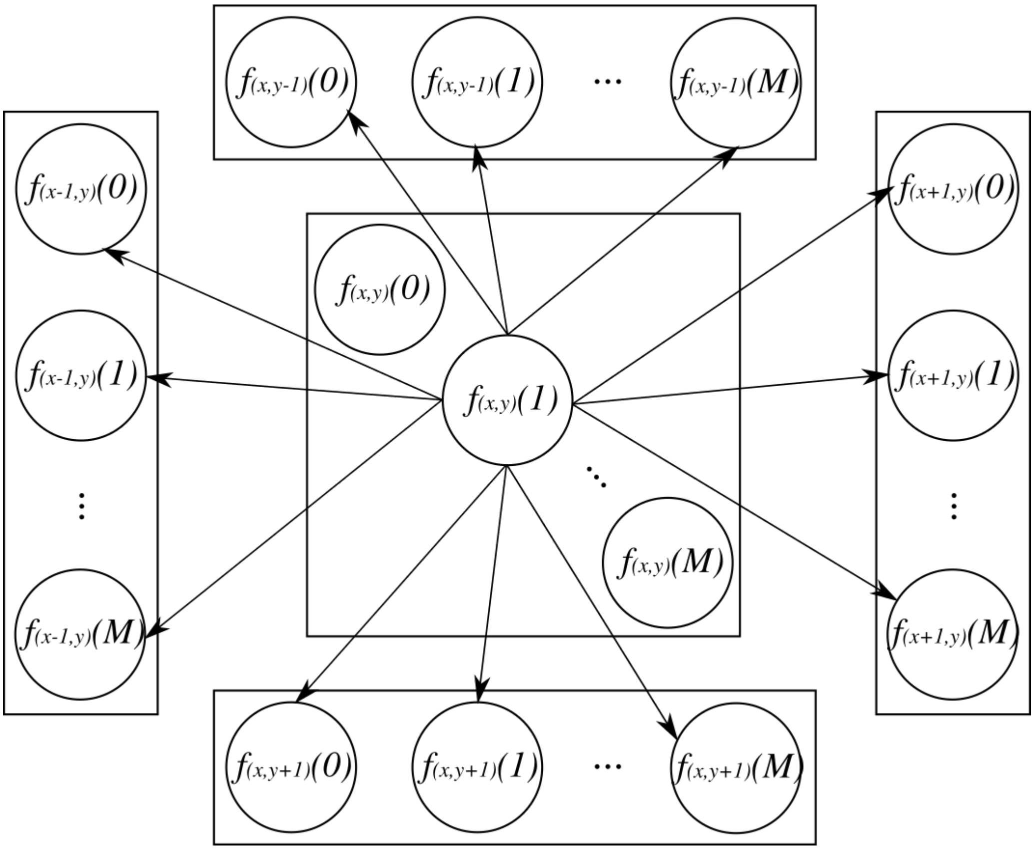


Рисунок 1.11 — Сусідні об'єкти до об'єкта (x, y). Вершини в об'єктах позначені окружностями, всередині яких зазначені штрафи за вибір міток. Задля наочності дужки показані лише для вершини з міткою $d = 1$

На дужку між вершиною з міткою $d \in D$ в об'єкті (x, y) і вершиною з міткою $d' \in D$ в об'єкті (x', y') накладається штраф за невідповідність обраних зсувів в сусідніх об'єктах $g_{(x,y),(x',y')}(d, d')$ (рис. 1.12). Множину всіх пар сусідніх об'єктів позначимо через $\mathcal{N}$. Таким чином, $g : \mathcal{N} \times D^2 \to \mathbb{R}$.

Відображення $\boldsymbol{d} : T \to D$ назвемо *розміткою*. Кожному пікселю зобра-

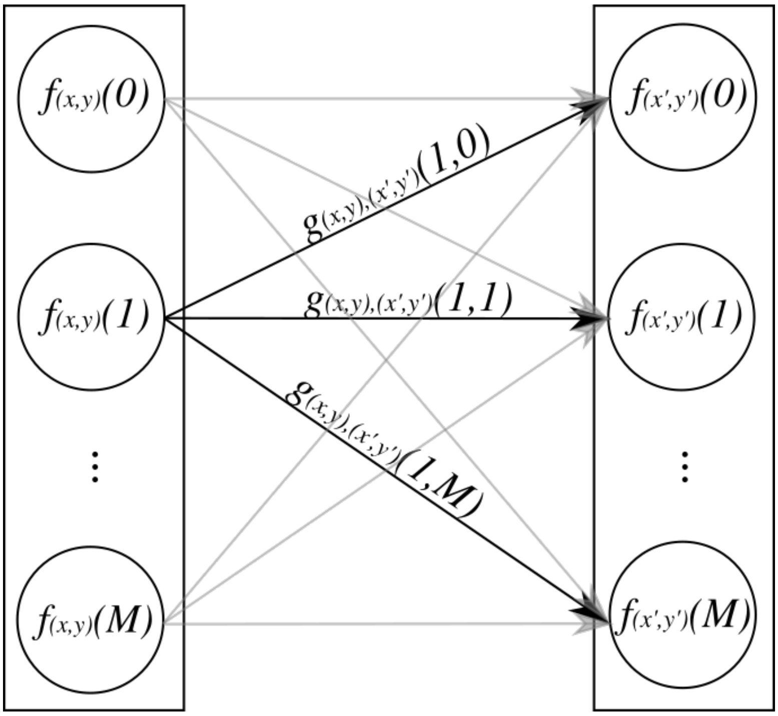


Рисунок 1.12 — Ваги дужок, що йдуть від вершини з міткою $d = 1$ в об'єкті (x, y) до всіх вершин в сусідньому об'єкті $(x', y') \in \mathcal{N}(x, y)$. Ваги інших дужок записуються аналогічно

ження (кожному об'єкту графу) воно ставить у відповідність мітку, тобто обирає одну й тільки одну вершину в кожному об'єкті (рис. 1.13). Задача полягає в такому виборі розмітки $\boldsymbol{d} \in D^T$, яка мінімізує штрафну функцію

$$\begin{aligned} G(\boldsymbol{d}) = &\sum_{(x,y)\in T} f_{(x,y)}(d(x,y)) + \\ &+ \sum_{((x,y),(x',y'))\in\mathcal{N}} g_{(x,y),(x',y')}(d(x,y), d(x',y')) . \end{aligned} \tag{1.5}$$

Висновки до розділу 1

Проведено огляд алгоритмів, які використовуються для розв'язання задачі стереобачення. Поставлена задача, розв'язання якої наведено в наступних

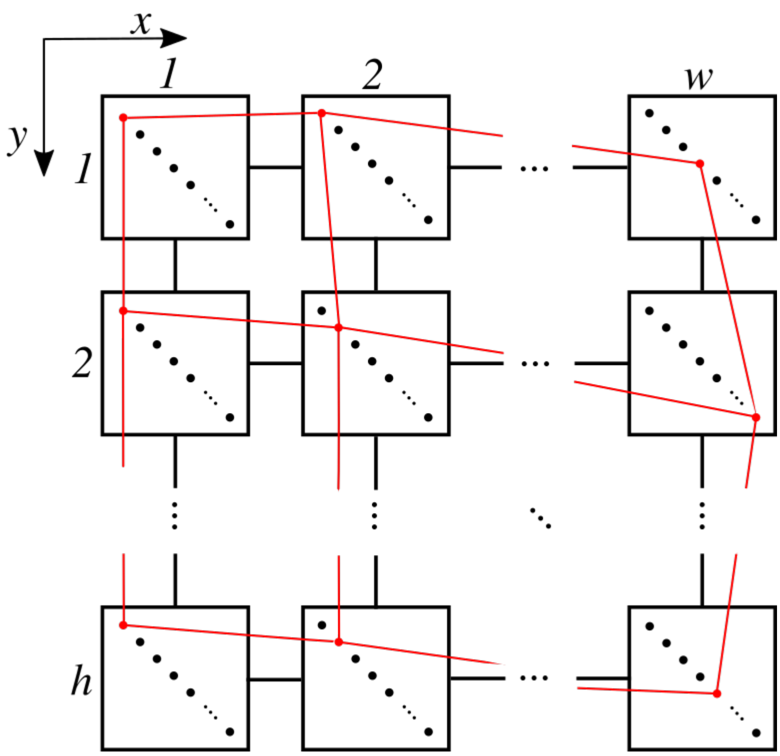


Рисунок 1.13 — Візуалізація розмітки. Об’єкти позначені квадратами. Об’єкти, поєднані чорними лініями є сусідніми. Вершини в об’єктах позначені кружечками. Розмітка позначена червоним

розділах дисертації.

Наведена постановка задачі є природною, адже враховує залежність проекції об’єктів з простору в одному пікселі від проекцій об’єктів в інших пікселях, наприклад, якщо один гладкий об’єкт зайняв більше одного рядка зображення. В постановці задачі, що використовується для розв’язання задачі стереобачення методом динамічного програмування, залежності між рядками немає, тобто вважається, що об’єкти, спроектовані в різні рядки зображення, є незалежними, що на практиці не відповідає дійсності.

2 ПОШУК КАРТИ ГЛИБИН ЗА СТЕРЕОПАРОЮ

Другий розділ присвячено розв'язку задачі стереобачення алгоритмом дифузії. Описуються властивості алгоритму та з'ясовується придатність розв'язку для поставленої задачі.

2.1 Відомості з теорії оптимізації

Введемо декілька визначень та тверджень [7], що знадобляться при розв'язанні задачі.

Означення. *Опуклим бакатогранником* (polyhedron) P в $\mathbb{R}^n$ називається множина, що може бути представлена скінченним набором лінійних нерівностей, тобто $P = \left\{ \boldsymbol{x} \in \mathbb{R}^n \;\middle|\; \hat{A} \cdot \boldsymbol{x} \leq \boldsymbol{b} \right\}$ для матриці $\hat{A} \in \mathbb{R}^{m \times n}$ і вектору $\boldsymbol{b} \in \mathbb{R}^n$. Обмежений опуклий багатогранник називається *політопом* (polytope).

Означення. Політоп

$$\Delta^n := \left\{ \boldsymbol{x} \in \mathbb{R}^n_+ \;\middle|\; \sum_{x=1}^{n} x_i = 1 \right\} \tag{2.1}$$

називається *n-вимірним (ймовірнісним) симплексом*.

Означення. Для множини $X \subset \mathbb{R}^n$ множина

$$conv\,(X) = \left\{ \boldsymbol{s} \in \mathbb{R}^n \;\middle|\; \exists N \in \mathbb{N}_+ \;:\; s = \sum_{i=1}^{N} p_i \cdot x^i,\, x_i \in X,\, \boldsymbol{p} \in \Delta^N \right\}$$

точок, які представлені як опукла комбінація скінченного числа точок з множини X, називається *опуклою оболонкою* множини X.

Означення. Нехай P — опуклий багатогранник в $\mathbb{R}^n$, визначений множиною лінійних обмежень. Оптимізаційні задачі виду

$$\min_{\boldsymbol{x}\in P}\langle \boldsymbol{c}, \boldsymbol{x}\rangle, \qquad \max_{\boldsymbol{x}\in P}\langle \boldsymbol{c}, \boldsymbol{x}\rangle$$

називаються *задачами лінійного програмування.*

Означення. Задача лінійного програмування з додатковим обмеженням, яке дозволяє всім змінним приймати значення тільки 0 або 1, наприклад,

$$\min_{\boldsymbol{x}\in P\cap\{0,1\}^n}\langle \boldsymbol{c}, \boldsymbol{x}\rangle,$$

називається *задачею булевого цілочисельного лінійного програмування.*

Означення. Нехай задача

$$\min_{\boldsymbol{x}\in X} f\left(\boldsymbol{x}\right) \tag{2.2}$$

є задачею мінімізації функції $f : X \to \mathbb{R}$ на множині $X \subseteq \mathbb{R}^n$. Тоді оптимізаційна задача

$$\min_{\boldsymbol{x}\in X'} g\left(\boldsymbol{x}\right)$$

називається *релаксацією* задачі (2.2), якщо $X' \supseteq X$ та $g\left(\boldsymbol{x}\right) \geq f\left(\boldsymbol{x}\right)$ для будь-якого $\boldsymbol{x} \in X$.

Означення. Для задачі (2.2) множина X називається *допустимою множиною*, а $\boldsymbol{x} \in X$ – *допустимою точкою.*

Означення. Розглянемо задачу цілочисельного лінійного програмування

$$\min_{\substack{\boldsymbol{x}\in P\cap\{0,1\}^n \\ \hat{A}\cdot\boldsymbol{x}=\boldsymbol{b}}}\langle \boldsymbol{c}, \boldsymbol{x}\rangle, \tag{2.3}$$

де P – політоп, $\hat{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^n$.

Дуальна функція Лагранжа для задачі (2.3) має вигляд

$$\min_{\boldsymbol{x} \in P \cap \{0,1\}^n} \langle \boldsymbol{c}, \boldsymbol{x} \rangle + \langle \boldsymbol{\lambda}, \hat{A} \cdot \boldsymbol{x} - \boldsymbol{b} \rangle, \qquad \boldsymbol{\lambda} \in \mathbb{R}^n. \tag{2.4}$$

Змінна $\boldsymbol{\lambda}$ називається *дуальною змінною*. Цей прийом називається *дуалізацією обмежень* $\hat{A} \cdot \boldsymbol{x} = \boldsymbol{b}$, а релаксації такого вигляду називаються *лагранжевими релаксаціями*. Наступна задача є дуальною задачею Лагранжа для задачі (2.3)

$$\max_{\boldsymbol{\lambda} \in \mathbb{R}^n} \min_{\boldsymbol{x} \in P \cap \{0,1\}^n} \langle \boldsymbol{c}, \boldsymbol{x} \rangle + \langle \boldsymbol{\lambda}, \hat{A} \cdot \boldsymbol{x} - \boldsymbol{b} \rangle. \tag{2.5}$$

Твердження. (Прямо-двоїсті умови оптимальності, Primal-dual optimality conditions) Вектор $\boldsymbol{\lambda}^*$ — це оптимум дуальної задачі (2.5) тоді й тільки тоді, коли існує

$$\begin{aligned} \boldsymbol{x}^* \in \arg &\min_{\boldsymbol{x} \in conv(P \cap \{0,1\}^n)} \langle \boldsymbol{c}, \boldsymbol{x} \rangle + \langle \boldsymbol{\lambda}, \hat{A} \cdot \boldsymbol{x} - \boldsymbol{b} \rangle = \\ &= \arg \min_{\boldsymbol{x} \in conv(P \cap \{0,1\}^n)} \langle \boldsymbol{c} + \hat{A}^T \cdot \boldsymbol{\lambda}, \boldsymbol{x} \rangle, \end{aligned} \tag{2.6}$$

такий що $\hat{A} \cdot \boldsymbol{x}^* = \boldsymbol{0}$. Іншими словами, $\boldsymbol{\lambda}^*$ — це дуальний оптимум тоді й тільки тоді, коли існує такий $\boldsymbol{x}^*$, що задовольняє (2.6) і є допустимим для релаксованої прямої задачі

$$\min_{\substack{\boldsymbol{x} \in conv(P \cap \{0,1\}^n) \\ \hat{A} \cdot \boldsymbol{x} = \boldsymbol{b}}} \langle \boldsymbol{c}, \boldsymbol{x} \rangle.$$

Такий $\boldsymbol{x}^*$ є розв’язком релаксованої прямої задачі.

Примітка. Якщо крім виконання умови (2.6) та $\hat{A} \cdot \boldsymbol{x}^* = \boldsymbol{0}$ справедливо, що $\boldsymbol{x}^* \in \{0,1\}^n$, тоді $\boldsymbol{x}^*$ — це розв’язок нерелаксованої цілочисельної задачі лінійного програмування (2.3).

2.2 Релаксація Лагранжа для поставленої оптимізаційної задачі

Введемо вектор $\boldsymbol{\mu}$, який містить змінну з множини $[0, 1]$ для кожної вершини $(x, y, d), (x, y) \in T, d \in D$ та кожної дужки $((x, y, d), (x', y', d'))$, що поєднує пари міток $d, d' \in D$ між сусідніми об'єктами $((x, y), (x', y')) \in \mathcal{N}$ в побудованому графі. Множину всіх вершин і дужок графу позначимо через $\mathcal{I}$.

Елементи даного вектору $\mu_{(x,y)}(d)$, що відповідають вершинам, мають бути узгодженими з розміткою $\boldsymbol{d}$, тобто якщо $\mu_{(x,y)}(d^*) = 1$ для якоїсь мітки $d^* \in D$, то $d(x, y) = d^*$. Розмітка $\boldsymbol{d}$, за визначенням, має в кожному об'єкті $(x, y) \in T$ обрати одну й тільки одну мітку $d \in D$. Таким чином, на елементи вектора $\boldsymbol{\mu}$ накладаються *обмеження однозначності для вершин* в об'єкті

$$\sum_{d \in D} \mu_{(x,y)}(d) = 1, \qquad \forall (x, y) \in T, \tag{2.7}$$

тобто у випадку, якщо вектор $\boldsymbol{\mu}$ — бінарний, то тільки один його елемент, що відповідає вершинам в об'єкті (x, y), може бути рівним одиниці.

Якщо в об'єкті обрана вершина d^*, то обрані дужки, що виходять із даного об'єкту до сусідніх об'єктів, мають виходити з цієї ж вершини (рис. 2.1). Ці обмеження називаються *поєднуючими*

$$\sum_{d \in D} \mu_{(x,y),(x',y')}(d, d') = \mu_{(x',y')}(d'), \qquad \forall ((x, y), (x',y')) \in \mathcal{N}, d' \in D. \tag{2.8}$$

З цих обмежень також випливає, що між двома сусідніми об'єктами може бути обрана лише одна дужка, тобто на вектор $\boldsymbol{\mu}$ ще накладаються *обмеження однозначності для дужок* між парами сусідніх об'єктів

$$\sum_{d, d' \in D} \mu_{(x,y),(x',y')}(d, d') = 1, \qquad \forall ((x, y), (x', y')) \in \mathcal{N}. \tag{2.9}$$

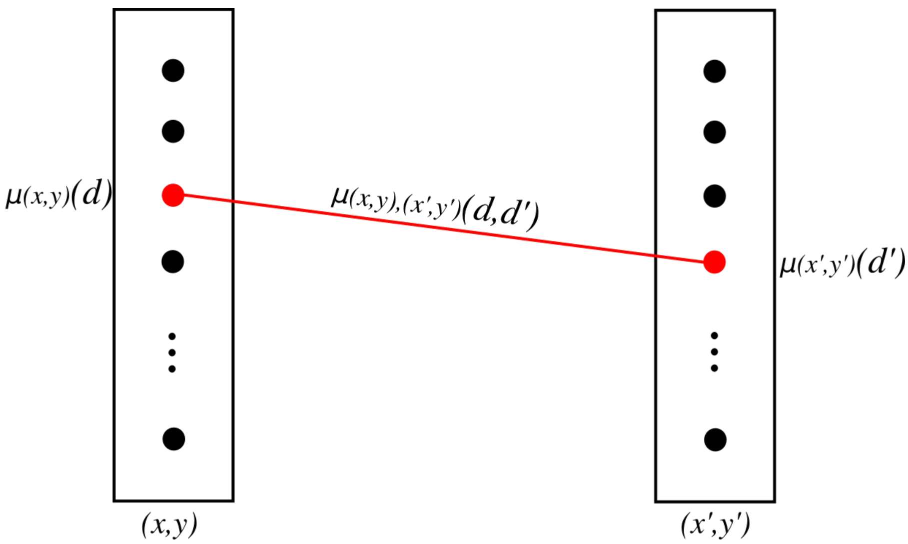


Рисунок 2.1 — Поєднуючі обмеження

Обмеження (2.7), (2.8) та (2.9) утворюють *локальний політоп*

$$\mathcal{L} := \begin{cases} \sum\limits_{d \in D} \mu_{(x,y),(x',y')}\left(d, d'\right) = \mu_{(x',y')}\left(d'\right), \ \forall \left(\left(x, y\right), \left(x',y'\right)\right) \in \mathcal{N}, d' \in D, \\ \sum\limits_{d \in D} \mu_{(x,y)}\left(d\right) = 1, \qquad \forall \left(x, y\right) \in T, \\ \sum\limits_{d, d' \in D} \mu_{(x,y),(x',y')}\left(d, d'\right) = 1, \qquad \forall \left(\left(x, y\right), \left(x', y'\right)\right) \in \mathcal{N}, \\ \boldsymbol{\mu} \geq \mathbf{0}, \end{cases} \tag{2.10}$$

де остання нерівність означає, що кожен елемент вектора $\boldsymbol{\mu}$ невід'ємний.

Отримаємо задачу цілочисельного лінійного програмування

$$\min_{\boldsymbol{d} \in D^T} E\left(\boldsymbol{d}\right) = \min_{\boldsymbol{\mu} \in \mathcal{L} \cap \{0,1\}^{\mathcal{I}}} \langle \boldsymbol{\theta}, \boldsymbol{\mu} \rangle,$$

де $\boldsymbol{\theta}$ — вектор штрафів, накладених на всі вершини та дуги, елементи якого розташовані в тій же послідовності, що й елементи вектора $\boldsymbol{\mu}$.

Випишемо скалярний добуток у правій частині останнього виразу в явному вигляді

$$\min_{\boldsymbol{\mu}\in\mathcal{L}\cap\{0,1\}^{\mathcal{I}}}\langle\boldsymbol{\theta},\boldsymbol{\mu}\rangle = \min_{\boldsymbol{\mu}\in\mathcal{L}\cap\{0,1\}^{\mathcal{I}}}\left[\sum_{(x,y)\in T}\sum_{d\in D}\mu_{(x,y)}(d)\cdot f_{(x,y)}(d) + \right. \\ \left. + \sum_{((x,y),(x',y'))\in\mathcal{N}}\sum_{d,d'\in D}\mu_{(x,y),(x',y')}(d,d')\cdot g_{(x,y),(x',y')}(d,d')\right]. \tag{2.11}$$

Використаємо прийом дуалізації (2.4) поєднуючих обмежень (2.8) в задачі (2.11). Новий доданок має вигляд

$$\sum_{(x,y)\in T}\sum_{(x',y')\in\mathcal{N}(x,y)}\sum_{d\in D}\varphi_{(x,y),(x',y')}(d)\cdot\left[\sum_{d'\in D}\mu_{(x,y),(x',y')}(d,d') - \mu_{(x,y)}(d)\right], \tag{2.12}$$

де змінні $\varphi_{(x,y),(x',y')}(d)\in\mathbb{R}$, $(x,y)\in T$, $(x',y')\in\mathcal{N}(x,y)$, $d\in D$, є дуальними. Будемо називати їх *потенціалами* (рис. 2.2).

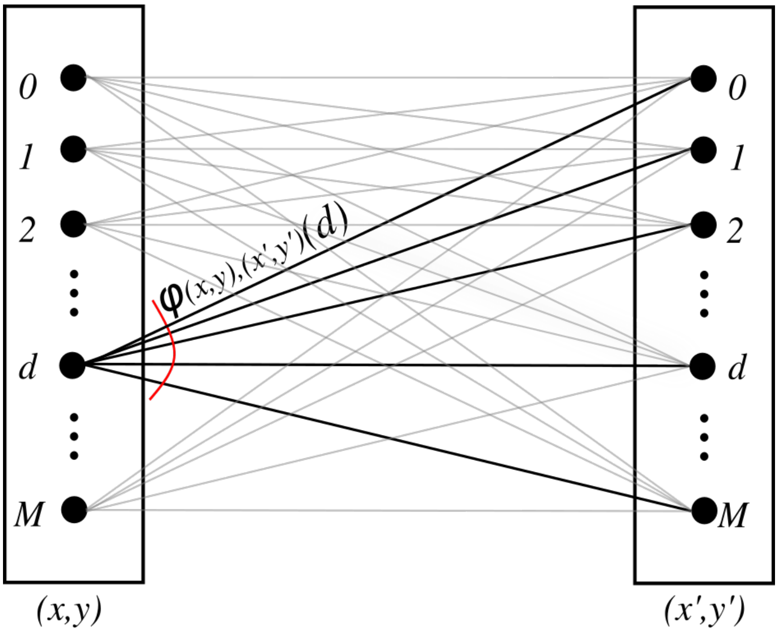


Рисунок 2.2 — Змінна $\varphi_{(x,y),(x',y')}(d)$

Запишемо цільову функцію задачі (2.11) з урахуванням нового доданку

(2.12), в якому розкриємо дужки

$$
\begin{aligned}
\langle \boldsymbol{\theta}^{\varphi}, \boldsymbol{\mu} \rangle = &\sum_{(x,y)\in T} \sum_{d\in D} \mu_{(x,y)}(d) \cdot f_{(x,y)}(d) + \\
&+ \sum_{((x,y),(x',y'))\in \mathcal{N}} \sum_{d,d'\in D} \mu_{(x,y),(x',y')}(d,d') \cdot g_{(x,y),(x',y')}(d,d') + \\
&+ \sum_{(x,y)\in T} \sum_{(x',y')\in \mathcal{N}(x,y)} \sum_{d\in D} \varphi_{(x,y),(x',y')}(d) \cdot \sum_{d'\in D} \mu_{(x,y),(x',y')}(d,d') - \\
&- \sum_{(x,y)\in T} \sum_{(x',y')\in \mathcal{N}(x,y)} \sum_{d\in D} \varphi_{(x,y),(x',y')}(d) \cdot \mu_{(x,y)}(d).
\end{aligned}
$$

Згрупуємо перший доданок з останнім, а другий — з третім

$$
\begin{aligned}
\langle \boldsymbol{\theta}^{\varphi}, \boldsymbol{\mu} \rangle = &\sum_{(x,y)\in T} \sum_{d\in D} \mu_{(x,y)}(d) \cdot \left[f_{(x,y)}(d) - \sum_{(x',y')\in \mathcal{N}(x,y)} \varphi_{(x,y),(x',y')}(d) \right] + \\
&+ \sum_{((x,y),(x',y'))\in \mathcal{N}} \sum_{d,d'\in D} \mu_{(x,y),(x',y')}(d,d') \cdot \big[g_{(x,y),(x',y')}(d,d') + \\
&+ \varphi_{(x,y),(x',y')}(d) + \varphi_{(x',y'),(x,y)}(d') \big].
\end{aligned}
\tag{2.13}
$$

Введемо позначення для *репараметризованого штрафу за вибір мітки* $d \in D$ в об'єкті $(x, y) \in T$

$$
f^{\varphi}_{(x,y)}(d) = f_{(x,y)}(d) - \sum_{(x',y')\in \mathcal{N}(x,y)} \varphi_{(x,y),(x',y')}(d), \tag{2.14}
$$

тобто репараметризований штраф у вершині отримується шляхом віднімання потенціалів, що виходять із даної вершини в усі сусідні об'єкти, від вихідного штрафу в вершині (рис. 2.3).

Також введемо позначення для *репараметризованого штрафу за вибір*

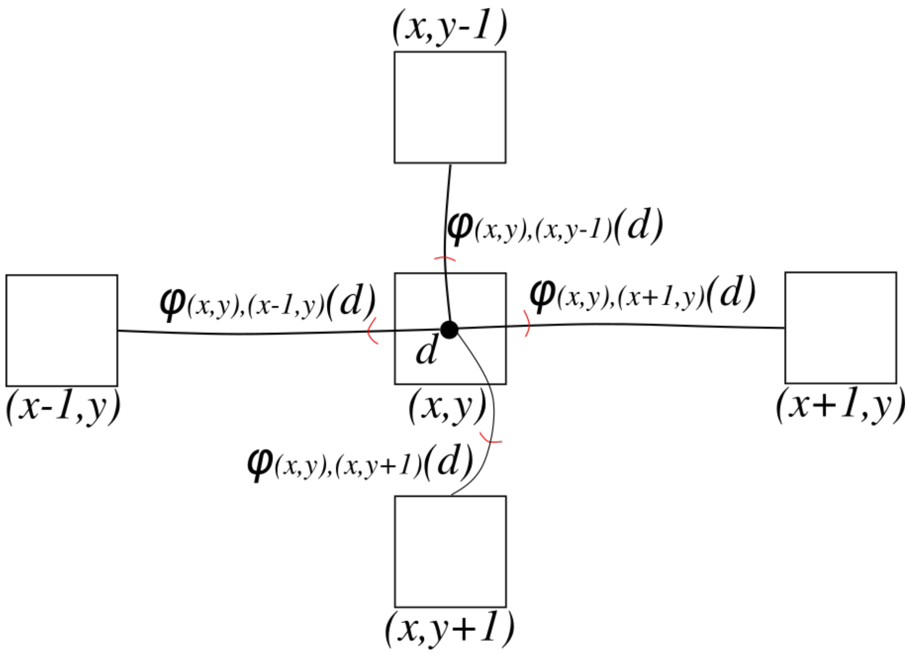


Рисунок 2.3 — Потенціали, які віднімаються від вихідного штрафу в вершині (x, y, d)

пари міток $d, d' \in D$ у двох сусідніх об'єктах $((x, y), (x', y')) \in \mathcal{N}$

$$g^{\varphi}_{(x,y),(x',y')}(d, d') = g_{(x,y),(x',y')}(d, d') + \varphi_{(x,y),(x',y')}(d) + \varphi_{(x',y'),(x,y)}(d'), \quad (2.15)$$

тобто репараметризований штраф на дужці отримується шляхом додавання потенціалів, що виходять в об'єкти, які дана дужка поєднує, до вихідного штрафу на дужці (рис. 2.4).

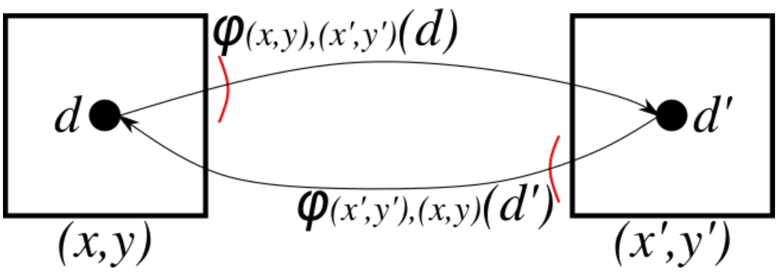


Рисунок 2.4 — Потенціали, які додаються до вихідного штрафу в дужці між вершиною (x, y, d) та вершиною (x', y', d')

Підставимо вирази (2.14) та (2.15) у (2.13)

$$\langle \boldsymbol{\theta}^{\varphi}, \boldsymbol{\mu} \rangle = \sum_{(x,y)\in T} \sum_{d\in D} \mu_{(x,y)}(d) \cdot f^{\varphi}_{(x,y)}(d) +$$
$$+ \sum_{((x,y),(x',y'))\in \mathcal{N}} \sum_{d,d'\in D} \mu_{(x,y),(x',y')}(d,d') \cdot g^{\varphi}_{(x,y),(x',y')}(d,d') .$$

Твердження. Після перетворень (2.14) та (2.14) значення штрафної функції не зміниться для будь-якої розмітки $\boldsymbol{d} \in D^T$.

Для доведення цього твердження запишемо штрафну функцію (1.5) з репараметризованими штрафами (2.14) та (2.14)

$$E^{\varphi}(\boldsymbol{d}) = \sum_{(x,y)\in T} f^{\varphi}_{(x,y)}(d(x,y)) +$$
$$+ \sum_{((x,y),(x',y'))\in \mathcal{N}} g^{\varphi}_{(x,y),(x',y')}(d(x,y), d(x',y')) =$$
$$= \sum_{(x,y)\in T} \left[f_{(x,y)}(d(x,y)) - \sum_{(x',y')\in \mathcal{N}(x,y)} \varphi_{(x,y),(x',y')}(d(x,y)) \right] +$$
$$+ \sum_{((x,y),(x',y'))\in \mathcal{N}} \left[g_{(x,y),(x',y')}(d(x,y), d(x',y')) + \right.$$
$$\left. + \varphi_{(x,y),(x',y')}(d(x,y)) + \varphi_{(x',y'),(x,y)}(d(x',y')) \right] .$$

Розкриємо дужки в останньому виразі

$$E^{\varphi}(\boldsymbol{d}) = \sum_{(x,y)\in T} f_{(x,y)}(d(x,y)) - \sum_{(x,y)\in T} \sum_{(x',y')\in \mathcal{N}(x,y)} \varphi_{(x,y),(x',y')}(d(x,y)) +$$
$$+ \sum_{((x,y),(x',y'))\in \mathcal{N}} g_{(x,y),(x',y')}(d(x,y), d(x',y')) +$$
$$+ \sum_{((x,y),(x',y'))\in \mathcal{N}} \left[\varphi_{(x,y),(x',y')}(d(x,y)) + \varphi_{(x',y'),(x,y)}(d(x',y')) \right] = E(\boldsymbol{d}) ,$$

адже перший та третій доданки дорівнюють $E(\boldsymbol{d})$, а інші доданки разом дорівнюють нулю. Отримали, що $E^{\varphi}(\boldsymbol{d}) = E(\boldsymbol{d})$ для всіх розміток $\boldsymbol{d} \in D^T$. Отже, репараметризовані ваги введено правильно.

Перепишемо локальний політоп в термінах симплексів (2.1)

$$\mathcal{L} = \begin{cases} \boldsymbol{\mu_{(x,y)}} \in \Delta^{T\times D}, \\ \boldsymbol{\mu_{(x,y),(x',y')}} \in \Delta^{\mathcal{N}\times D^2}, \\ \sum\limits_{d'\in D} \mu_{(x,y),(x',y')}(d,d') = \mu_{(x,y)}(d), \; \forall (x,y) \in T, (x',y') \in \mathcal{N}(x,y), d \in D. \end{cases}$$

Тоді

$$\min_{\boldsymbol{\mu}\in\mathcal{L}\cap\{0,1\}^{\mathcal{I}}} \langle \boldsymbol{\theta}^{\varphi}, \boldsymbol{\mu} \rangle = \min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}} \\ \boldsymbol{\mu_{(x,y)}}\in\Delta^{T\times D} \\ \boldsymbol{\mu_{(x,y),(x',y')}}\in\Delta^{\mathcal{N}\times D^2}}} \langle \boldsymbol{\theta}^{\varphi}, \boldsymbol{\mu} \rangle.$$

Маємо двоїсту задачу Лагранжа (2.5) для вихідної оптимізаційної задачі, що полягає в мінімізації енергії (1.5),

$$\max_{\Phi} \min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}} \\ \boldsymbol{\mu_{(x,y)}}\in\Delta^{T\times D} \\ \boldsymbol{\mu_{(x,y),(x',y')}}\in\Delta^{\mathcal{N}\times D^2}}} \langle \boldsymbol{\theta}^{\varphi}, \boldsymbol{\mu} \rangle,$$

де $\Phi = \left\{ \varphi_{(x,y),(x',y')}(d) \in \mathbb{R} \;\middle|\; (x,y) \in T,\, (x',y') \in \mathcal{N}(x,y),\, d \in D \right\}$ — множина всіх потенціалів. Розпишемо вираз, який максимізується, для вершин і дужок

$$\min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}} \\ \boldsymbol{\mu_{(x,y)}}\in\Delta^{T\times D} \\ \boldsymbol{\mu_{(x,y),(x',y')}}\in\Delta^{\mathcal{N}\times D^2}}} \langle \boldsymbol{\theta}^{\varphi}, \boldsymbol{\mu} \rangle =$$

$$+ \min_{\boldsymbol{\mu_{(x,y)}}\in\Delta^{T\times D}\cap\{0,1\}^{T\times D}} \sum_{(x,y)\in T} \sum_{d\in D} \mu_{(x,y)}(d) \cdot f^{\varphi}_{(x,y)}(d) +$$

$$+ \min_{\boldsymbol{\mu_{(x,y),(x',y')}}\in\Delta^{\mathcal{N}\times D^2}\cap\{0,1\}^{\mathcal{N}\times D^2}} \sum_{((x,y),(x',y'))\in\mathcal{N}} \sum_{d,d'\in D} \mu_{(x,y),(x',y')}(d,d') \cdot g_{(x,y),(x',y')}(d,d').$$

Використаємо обмеження однозначності (2.7) та (2.9)

$$\min_{\substack{\boldsymbol{\mu} \in \{0,1\}^{\mathcal{I}} \\ \boldsymbol{\mu}_{(\boldsymbol{x},\boldsymbol{y})} \in \Delta^{T \times D} \\ \boldsymbol{\mu}_{(\boldsymbol{x},\boldsymbol{y}),(\boldsymbol{x}',\boldsymbol{y}')} \in \Delta^{\mathcal{N} \times D^2}}} \langle \boldsymbol{\theta}^{\varphi}, \boldsymbol{\mu} \rangle = \\ = \sum_{(x,y) \in T} \min_{d \in D} f^{\varphi}_{(x,y)}(d) + \sum_{((x,y),(x',y')) \in \mathcal{N}} \min_{d,d' \in D} g^{\varphi}_{(x,y),(x',y')}(d,d') \,. \tag{2.16}$$

Отримали остаточний вигляд дуальної функції Лагранжа, яку будемо максимізувати по набору двоїстих змінних Φ за допомогою алгоритма дифузії.

2.3 Алгоритм дифузії для розв’язання задачі стереобачення

Алгоритм дифузії є блочно-координатним підйомом [7], тобто на кожній ітерації відбувається максимізація цільової функції (2.16) за блоком змінних. При цьому змінні, що не входять в даних блок, залишаються фіксованими з попередньої ітерації.

Виведемо алгоритм дифузії, ітерація якого буде складатися з двох кроків. Виразимо дуальну змінну $\varphi^{i+1}_{(x,y),(x',y')}(d)$ на кроці $i+1$ через дуальну змінну $\varphi^{i}_{(x,y),(x',y')}(d)$ на кроці i. Зауважимо, що всі інші дуальні змінні не впливають на оптимізацію, бо не залежать від даної вершини, тому вважаємо, що вони дорівнюють нулю. Хочемо, щоб після перетворення мінімальні штрафи за вибір дужок, які йдуть від вершини (x, y, d) до сусідніх об’єктів $(x', y') \in \mathcal{N}(x, y)$, стали рівними нулю

$$\min_{d' \in D} g^{\varphi^{i+1}}_{(x,y),(x',y')}(d, d') = 0, \, \forall t' \in \mathcal{N}(t)\,, \, d \in D. \tag{2.17}$$

Підставимо репараметризований штраф (2.15)

$$\min_{d'\in D} g^{\varphi^{i+1}}_{(x,y),(x',y')}\left(d,d'\right) =$$
$$= \min_{d'\in D}\left[g_{(x,y),(x',y')}\left(d,d'\right) + \varphi^{i+1}_{(x,y),(x',y')}\left(d\right) + \varphi^{i+1}_{(x',y'),(x,y)}\left(d'\right)\right] = 0.$$

Як уже зазначалося, вважаємо, що $\varphi^{i+1}_{(x',y'),(x,y)}\left(d'\right) = 0$. Виразимо $g_{(x,y),(x',y')}\left(d,d'\right)$ через штраф на кроці i

$$\min_{d'\in D}\left[g^{i}_{(x,y),(x',y')}\left(d,d'\right) - \varphi^{i}_{(x,y),(x',y')}\left(d\right) + \varphi^{i+1}_{(x,y),(x',y')}\left(d\right)\right] = 0.$$

Два останні доданки не залежать від змінної $d' \in D$, по якій йде мінімізація, отже, винесемо їх за знак мінімуму

$$\min_{d'\in D} g^{i}_{(x,y),(x',y')}\left(d,d'\right) - \varphi^{i}_{(x,y),(x',y')}\left(d\right) + \varphi^{i+1}_{(x,y),(x',y')}\left(d\right) = 0.$$

Звідси маємо вираз для *першого кроку алгоритму дифузії*

$$\varphi^{i+1}_{(x,y),(x',y')}\left(d\right) = \varphi^{i}_{(x,y),(x',y')}\left(d\right) - \min_{d'\in D} g^{i}_{(x,y),(x',y')}\left(d,d'\right).$$

Виразимо дуальну змінну $\varphi^{i+2}_{(x,y),(x',y')}\left(d\right)$ на кроці $i+2$ через дуальну змінну $\varphi^{i+1}_{(x,y),(x',y')}\left(d\right)$ на кроці $i+1$. Хочемо, щоб після перетворення штраф за вибір вершини (x,y,d) розподілився рівномірно між дужками, що йдуть від цієї вершини до сусідніх об'єктів, тобто щоб виконувалась рівність

$$\min_{d'\in D} g^{\varphi^{i+2}}_{(x,y),(x',y')}\left(d,d'\right) = \frac{f^{i+1}_{(x,y)}\left(d\right)}{\left|\mathcal{N}\left(x,y\right)\right|},\ \forall t' \in \mathcal{N}\left(t\right),\ d \in D.$$

Знову розпишемо ліву частину рівності

$$\min_{d'\in D}\left[g_{(x,y),(x',y')}\left(d,d'\right)+\varphi^{i+2}_{(x,y),(x',y')}\left(d\right)\right]=\frac{f^{i+1}_{(x,y)}\left(d\right)}{\left|\mathcal{N}\left(x,y\right)\right|}.$$

Звідси

$$\varphi^{i+2}_{(x,y),(x',y')}\left(d\right)=\frac{f^{i+1}_{(x,y)}\left(d\right)}{\left|\mathcal{N}\left(x,y\right)\right|}-\min_{d'\in D}g_{(x,y),(x',y')}\left(d,d'\right)=$$
$$=\frac{f^{i+1}_{(x,y)}\left(d\right)}{\left|\mathcal{N}\left(x,y\right)\right|}-\min_{d'\in D}\left[g^{i+1}_{(x,y),(x',y')}\left(d,d'\right)-\varphi^{i+1}_{(x,y),(x',y')}\left(d\right)\right]=$$
$$=\varphi^{i+2}_{(x,y),(x',y')}\left(d\right)-\min_{d'\in D}g^{i+1}_{(x,y),(x',y')}\left(d,d'\right)+\frac{f^{i+1}_{(x,y)}\left(d\right)}{\left|\mathcal{N}\left(x,y\right)\right|}.$$

Використаємо рівність нулю мінімального штрафу за вибір дужки на попередньому кроці (2.17) та отримаємо вираз для *другого кроку алгоритму дифузії*

$$\varphi^{i+2}_{(x,y),(x',y')}\left(d\right)=\varphi^{i+2}_{(x,y),(x',y')}\left(d\right)-\frac{f^{i+1}_{(x,y)}\left(d\right)}{\left|\mathcal{N}\left(x,y\right)\right|}.$$

Таким чином, *елементарний крок алгоритм дифузії* 2.5 складається з двох операцій для кожного об'єкту (y,x):

$$\begin{gathered}\forall\left(x',y'\right)\in\mathcal{N}\left(x,y\right)\quad\forall d\in D\\ \varphi^{i+1}_{(x,y),(x',y')}\left(d\right)=\varphi^{i}_{(x,y),(x',y')}\left(d\right)-\min_{d'\in D}g^{\varphi^{i}}\left(d,d'\right),\end{gathered}\tag{2.18}$$

та

$$\begin{gathered}\forall\left(x',y'\right)\in\mathcal{N}\left(x,y\right)\quad\forall d\in D\\ \varphi^{i+2}_{(x,y),(x',y')}\left(d\right)=\varphi^{i+1}_{(x,y),(x',y')}\left(d\right)+\frac{f^{\varphi^{i+1}}_{(x,y)}\left(d\right)}{\left|\mathcal{N}\left(x,y\right)\right|},\end{gathered}\tag{2.19}$$

де через i позначено номер кроку.

Ітерація алгоритму дифузії полягає у виконанні кроків (2.18) та (2.19) для всіх об'єктів $(x,y)\in T$. На першій ітерації вважається, що всі дуальні змінні дорівнюють нулю, $\varphi_{(x,y),(x',y')}\left(d\right)=0$ для всіх об'єктів (x,y), для всіх сусідніх об'єктів $(x', y') \in \mathcal{N}(x, y)$ та для всіх міток $d \in D$.

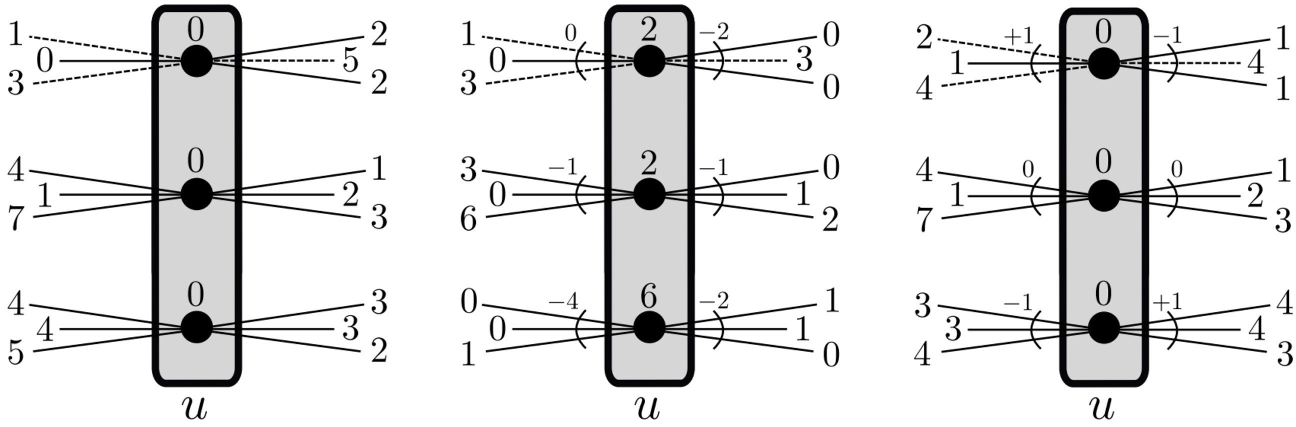


Рисунок 2.5 — Приклад виконання елементарного кроку дифузії для вершин в об'єкті u. На ілюстрації кожна з вершин має по два сусіди. Значення репараметризованих штрафів позначені великими числами, а значення дуальних змінних φ — меншими. Пунктирні лінії позначають мінімальні штрафи за вибір дужки від конкретної вершини до кожного сусіднього об'єкта. Зліва зображені початкові штрафи, посередині — штрафи та значення двоїстих змінних після виконання кроку (2.18), справа — після виконання кроку (2.19). Зображення взято з монографії [7]

2.3.1 Властивості алгоритму дифузії

Покажемо, що після виконання кожного з перетворень (2.18) та (2.19), що утворюють елементарний крок алгоритму дифузії, значення дуальної функції (2.16) не зменшується.

Твердження. Перший крок алгоритму дифузії (2.18) не зменшує значення дуальної функції Лагранжа (2.16).

Для доведення цього твердження розпишемо цільову функцію задачі

(2.16) після виконання кроку $i+1$

$$\min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}}\\ \boldsymbol{\mu_{(x,y)}}\in\Delta^{T\times D}\\ \boldsymbol{\mu_{(x,y),(x',y')}}\in\Delta^{\mathcal{N}\times D^2}}} \langle \boldsymbol{\theta}^{\varphi^{i+1}}, \boldsymbol{\mu}\rangle =$$

$$= \sum_{(x,y)\in T} \min_{d\in D} f^{\varphi^{i+1}}_{(x,y)}(d) + \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d,d'\in D} g^{\varphi^{i+1}}_{(x,y),(x',y')}(d,d') =$$

Підставимо вирази для репараметризованих штрафів (2.14) та (2.15)

$$= \sum_{(x,y)\in T} \min_{d\in D} \left[f_{(x,y)}(d) - \sum_{(x',y')\in\mathcal{N}(x,y)} \varphi^{i+1}_{(x,y),(x',y')}(d) \right] +$$
$$+ \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d,d'\in D} \left[g_{(x,y),(x',y')}(d,d') + \varphi^{i+1}_{(x,y),(x',y')}(d) + \varphi^{i+1}_{(x',y'),(x,y)}(d') \right] =$$

Підставимо вирази для оновлення дуальних змінних на кроці $i+1$ через дуальні змінні на кроці i (2.18)

$$= \sum_{(x,y)\in T} \min_{d\in D} \left\{ f_{(x,y)}(d) - \sum_{(x',y')\in\mathcal{N}(x,y)} \left[\varphi^{i}_{(x,y),(x',y')}(d) - \min_{d'\in D} g^{\varphi^i}_{(x,y),(x',y')}(d,d') \right] \right\} +$$
$$+ \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d,d'\in D} \Big[\quad g_{(x,y),(x',y')}(d,d') +$$
$$+\varphi^{i}_{(x,y),(x',y')}(d) - \min_{d''\in D} g^{\varphi^i}_{(x,y),(x',y')}(d,d'') +$$
$$+\varphi^{i}_{(x',y'),(x,y)}(d') - \min_{d''\in D} g^{\varphi^i}_{(x',y'),(x,y)}(d',d'') \Big] =$$

Внесемо сумму в дужки в першому доданку, а в другому застосуємо формулу для репараметризованого штрафу за вибір пари міток на кроці i (2.15)

$$= \sum_{(x,y)\in T} \min_{d\in D} \left[f_{(x,y)}\left(d\right) - \sum_{(x',y')\in\mathcal{N}(x,y)} \varphi^i_{(x,y),(x',y')}\left(d\right) + \right.$$

$$\left. + \sum_{(x',y')\in\mathcal{N}(x,y)} \min_{d'\in D} g^{\varphi^i}_{(x,y),(x',y')}\left(d,d'\right) \right] +$$

$$+ \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d,d'\in D} \left[g^{\varphi^i}_{(x,y),(x',y')}\left(d,d'\right) - \min_{d''\in D} g^{\varphi^i}_{(x,y),(x',y')}\left(d,d''\right) - \right.$$

$$\left. - \min_{d''\in D} g^{\varphi^i}_{(x',y'),(x,y)}\left(d',d''\right) \right] \geq$$

В першому доданку застосуємо формулу для репараметризованого штрафу за вибір вершини з міткою d в об'єкті (x, y) на кроці i (2.14), а в другому доданку скористаємось тим фактом, що сума мінімумів функцій не перевищує мінімуму суми функцій

$$\geq \sum_{(x,y)\in T} \sum_{d\in D} \left[f^{\varphi^i}_{(x,y)}\left(d\right) + \sum_{(x',y')\in\mathcal{N}(x,y)} \min_{d'\in D} g^{\varphi^i}_{(x,y),(x',y')}\left(d,d'\right) \right] +$$

$$+ \sum_{((x,y),(x',y'))\in\mathcal{N}} \left[\min_{d,d'\in D} g^{\varphi^i}_{(x,y),(x',y')}\left(d,d'\right) - \min_{d,d'\in D} g^{\varphi^i}_{(x,y),(x',y')}\left(d,d'\right) - \right.$$

$$\left. - \min_{d,d'\in D} g^{\varphi^i}_{(x',y'),(x,y)}\left(d',d\right) \right] \geq$$

Тепер у першому доданку скористаємось тим фактом, що сума мінімумів функцій не перевищує мінімуму суми функцій. Також зазначимо, що через те, що дуги в графі не направлені, має місце рівність

$$g^{\varphi^i}_{(x,y),(x',y')}\left(d,d'\right) = g^{\varphi^i}_{(x',y'),(x,y)}\left(d',d\right),$$

тому ці штрафи знищуються в другому доданку

$$\geq \sum_{(x,y)\in T}\left[\min_{d\in D} f^{\varphi^i}_{(x,y)}(d) + \min_{d\in D}\sum_{(x',y')\in\mathcal{N}(x,y)}\min_{d'\in D} g^{\varphi^i}_{(x,y),(x',y')}(d,d')\right] -$$
$$- \sum_{((x,y),(x',y'))\in\mathcal{N}}\min_{d,d'\in D} g^{\varphi^i}_{(x,y),(x',y')}(d,d') \geq$$

В першому доданку поміняємо місцями знак суми по сусіднім об'єктам $(x',y')\in\mathcal{N}(x,y)$ та знак мінімума по міткам $d\in D$ та розкриємо дужки

$$\geq \sum_{(x,y)\in T}\min_{d\in D} f^{\varphi^i}_{(x,y)}(d) + \sum_{(x,y)\in T}\sum_{(x',y')\in\mathcal{N}(x,y)}\min_{d,d'\in D} g_{(x,y),(x',y')}(d,d') -$$
$$- \sum_{((x,y),(x',y'))\in\mathcal{N}}\min_{d,d'\in D} g_{(x,y),(x',y')}(d,d') =$$

Другий доданок містить кожну дужку $((x,y),(x',y'))$ два рази, бо об'єкт (x',y') є сусіднім для об'єкта (x,y), а об'єкт (x,y) в свою чергу є сусіднім для об'єкта (x',y')

$$= \sum_{(x,y)\in T}\min_{d\in D} f^{\varphi^i}_{(x,y)}(d) + \sum_{((x,y),(x',y'))\in\mathcal{N}}\min_{d,d'\in D} g_{(x,y),(x',y')}(d,d') =$$
$$= \min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}}\\ \boldsymbol{\mu}_{(x,y)}\in\Delta^{T\times D}\\ \boldsymbol{\mu}_{(x,y),(x',y')}\in\Delta^{\mathcal{N}\times D^2}}} \langle\boldsymbol{\theta}^{\varphi^i},\boldsymbol{\mu}\rangle.$$

Отримали нерівність

$$\min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}}\\ \boldsymbol{\mu}_{(x,y)}\in\Delta^{T\times D}\\ \boldsymbol{\mu}_{(x,y),(x',y')}\in\Delta^{\mathcal{N}\times D^2}}} \langle\boldsymbol{\theta}^{\varphi^{i+1}},\boldsymbol{\mu}\rangle \geq \min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}}\\ \boldsymbol{\mu}_{(x,y)}\in\Delta^{T\times D}\\ \boldsymbol{\mu}_{(x,y),(x',y')}\in\Delta^{\mathcal{N}\times D^2}}} \langle\boldsymbol{\theta}^{\varphi^i},\boldsymbol{\mu}\rangle,$$

тобто крок (2.18) алгоритму дифузії не зменшує дуальную функцію Лагранжа (2.16).

Твердження. Другий крок алгоритму дифузії (2.19) не зменшує значення дуальної функції Лагранжа (2.16).

Для доведення цього твердження розпишемо цільову функцію задачі (2.16) після виконання кроку $i+2$

$$\min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}}\\ \boldsymbol{\mu}_{(x,y)}\in\Delta^{T\times D}\\ \boldsymbol{\mu}_{(x,y),(x',y')}\in\Delta^{\mathcal{N}\times D^2}}} \langle \boldsymbol{\theta}^{\varphi^{i+2}}, \boldsymbol{\mu}\rangle =$$

$$= \sum_{(x,y)\in T} \min_{d\in D} f^{\varphi^{i+2}}_{(x,y)}(d) + \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d,d'\in D} g^{\varphi^{i+2}}_{(x,y),(x',y')}(d,d') =$$

Підставимо вирази для репараметризованих штрафів (2.14) та (2.15)

$$= \sum_{(x,y)\in T} \min_{d\in D} \left[f_{(x,y)}(d) - \sum_{(x',y')\in\mathcal{N}(x,y)} \varphi^{i+1}_{(x,y),(x',y')}(d) \right] +$$
$$+ \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d,d'\in D} \left[g_{(x,y),(x',y')}(d,d') + \varphi^{i+2}_{(x,y),(x',y')}(d) + \varphi^{i+2}_{(x',y'),(x,y)}(d') \right] =$$

Підставимо вирази для оновлення потенціалів φ на кроці $i{+}2$ через потенціали на кроці $i+1$ (2.19)

$$= \sum_{(x,y)\in T} \min_{d\in D} \left\{ f_{(x,y)}(d) - \sum_{(x',y')\in\mathcal{N}(x,y)} \left[\varphi^{i+1}_{(x,y),(x',y')}(d) + \frac{f^{\varphi^{i+1}}_{(x,y)}(d)}{|\mathcal{N}(x,y)|} \right] \right\} +$$
$$+ \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d,d'\in D} \left[g_{(x,y),(x',y')}(d,d') + \varphi^{i+1}_{(x,y),(x',y')}(d) + \frac{f^{\varphi^{i+1}}_{(x,y)}(d)}{|\mathcal{N}(x,y)|} + \right.$$
$$\left. + \varphi^{i+1}_{(x',y'),(x,y)}(d') + \frac{f^{\varphi^{i+1}}_{(x',y')}(d')}{|\mathcal{N}(x',y')|} \right] =$$

В першому доданку розкриємо дужки, а в другому застосуємо формулу для репараметризованого штрафу за вибір пари міток на кроці $i+1$ (2.15)

$$= \sum_{(x,y)\in T} \min_{d\in D} \left[f_{(x,y)}(d) - \sum_{(x',y')\in\mathcal{N}(x,y)} \varphi^{i+1}_{(x,y),(x',y')}(d) - \sum_{(x',y')\in\mathcal{N}(x,y)} \frac{f^{\varphi^{i+1}}_{(x,y)}(d)}{|\mathcal{N}(x,y)|} \right] +$$

$$+ \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d,d'\in D} \left[g^{\varphi^{i+1}}_{(x,y),(x',y')}(d,d') + \frac{f^{\varphi^{i+1}}_{(x,y)}(d)}{|\mathcal{N}(x,y)|} + \frac{f^{\varphi^{i+1}}_{(x',y')}(d')}{|\mathcal{N}(x',y')|} \right] \geq$$

В першому доданку використаємо формулу для репараметризованого штрафу за вибір мітки на кроці $i+1$ (2.15), а в другому доданку внесемо знак мінімума в дужки

$$\geq \sum_{(x,y)\in T} \min_{d\in D} \left[f^{\varphi^{i+1}}_{(x,y)}(d) - \sum_{(x',y')\in\mathcal{N}(x,y)} \frac{f^{\varphi^{i+1}}_{(x,y)}}{|\mathcal{N}(x.y)|} \right] +$$

$$+ \sum_{((x,y),(x',y'))\in\mathcal{N}} \left[\min_{d,d'\in D} g^{\varphi^{i+1}}_{(x,y),(x',y')}(d,d') + \min_{d\in D} \frac{f^{\varphi^{i+1}}_{(x,y)}(d)}{|\mathcal{N}(x,y)|} + \min_{d'\in D} \frac{f^{\varphi^{i+1}}_{(x',y')}(d')}{|\mathcal{N}(x',y')|} \right] \geq$$

Розкриємо дужки в обох доданках

$$\geq \sum_{(x,y)\in T} \min_{d\in D} f^{\varphi^{i+1}}_{(x,y)}(d) - \sum_{(x,y)\in T} \min_{d\in D} \sum_{(x',y')\in\mathcal{N}(x,y)} \frac{f^{\varphi^{i+1}}_{(x,y)}(d)}{|\mathcal{N}(x,y)|} +$$

$$+ \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d,d'\in D} g^{\varphi^{i+1}}_{(x,y),(x',y')}(d,d') + \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d\in D} \frac{f^{\varphi^{i+1}}_{(x,y)}(d)}{|\mathcal{N}(x,y)|} +$$

$$+ \sum_{((x,y),(x',y'))\in\mathcal{N}} \min_{d'\in D} \frac{f^{\varphi^{i+1}}_{(x',y')}(d')}{|\mathcal{N}(x',y')|} =$$

Два останні доданки рівні між собою, а в сумі вони дорівнюють другому доданку за знаком «мінус», отже, вони знищуються. Те, що залишилося, дорівнює дуальній функції Лагранжа (2.16) на кроці $i+1$. Отримали нерівність

$$\min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}}\\ \boldsymbol{\mu}_{(\boldsymbol{x},\boldsymbol{y})}\in\Delta^{T\times D}\\ \boldsymbol{\mu}_{(\boldsymbol{x},\boldsymbol{y}),(\boldsymbol{x}',\boldsymbol{y}')}\in\Delta^{\mathcal{N}\times D^2}}} \langle \boldsymbol{\theta}^{\varphi^{i+2}}, \boldsymbol{\mu}\rangle \geq \min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}}\\ \boldsymbol{\mu}_{(\boldsymbol{x},\boldsymbol{y})}\in\Delta^{T\times D}\\ \boldsymbol{\mu}_{(\boldsymbol{x},\boldsymbol{y}),(\boldsymbol{x}',\boldsymbol{y}')}\in\Delta^{\mathcal{N}\times D^2}}} \langle \boldsymbol{\theta}^{\varphi^{i+1}}, \boldsymbol{\mu}\rangle,$$

тобто крок (2.19) алгоритму дифузії не зменшує дуальную функцію Лагранжа (2.16).

Таким чином, операції 2.18 і 2.19 максимізують дуальну функцію Лагранжа (2.16) по потенціалам Φ, а тому мінімізують штрафну функцію 1.5 [7], що випливає з прямо-двоїстих умов оптимальності.

Алгоритм полягає в ітеративному повторі елементарного кроку дифузії для всіх об'єктів $(x, y) \in T$ до виконання критерію зупинки. Прикладом критерію зупинки алгоритму може бути мале збільшення значення дуальної функції Лагранжа (2.16)

$$\min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}}\\ \boldsymbol{\mu}_{(\boldsymbol{x},\boldsymbol{y})}\in\Delta^{T\times D}\\ \boldsymbol{\mu}_{(\boldsymbol{x},\boldsymbol{y}),(\boldsymbol{x}',\boldsymbol{y}')}\in\Delta^{\mathcal{N}\times D^2}}} \langle \boldsymbol{\theta}^{\varphi^{i+1}}, \boldsymbol{\mu}\rangle - \min_{\substack{\boldsymbol{\mu}\in\{0,1\}^{\mathcal{I}}\\ \boldsymbol{\mu}_{(\boldsymbol{x},\boldsymbol{y})}\in\Delta^{T\times D}\\ \boldsymbol{\mu}_{(\boldsymbol{x},\boldsymbol{y}),(\boldsymbol{x}',\boldsymbol{y}')}\in\Delta^{\mathcal{N}\times D^2}}} \langle \boldsymbol{\theta}^{\varphi^{i}}, \boldsymbol{\mu}\rangle \leq \varepsilon,$$

де $\varepsilon \in \mathbb{R}_+$ — мале число. Виконання цього критерію означає, що ми наблизилися до шуканого максимуму дуальної функції Лагранжа або потрапили на нерухому точку алгоритму.

2.4 Вибір найкращої розмітки

Після мінімізації енергії задачі (1.5) треба знайти одну з тих розміток, штраф яких дорівнює мінімальному штрафу. Для цього використовується *алгоритм викреслювання другого порядку* (relaxation labeling algorithm) [7].

Розглядається той самий $|T|$-дольний граф, в кожній долі (об'єкті) якого міститься $|D|$ вершин. Вага вершини з міткою $d \in D$ в об'єкті $(x, y) \in T$ тепер дорівнює $\mu_{(x,y)}(d) \in \{0, 1\}$, а вага дужки між парою міток $d \in D$ в об'єкті $(x, y) \in T$ та $d' \in D$ в об'єкті $(x', y') \in \mathcal{N}(x, y) - \mu_{(x,y),(x',y')}(d, d')$.

Вершина (x, y, d) вважається *допустимою*, якщо відповідна їй змінна $\mu_{(x,y)}(d) = 1$. Аналогічно, пара вершин $((x, y, d), (x', y', d'))$ вважається *допустимою*, якщо відповідна змінна $\mu_{(x,y),(x',y')}(d, d') = 1$.

На побудованому графі розв'язується $(\bigvee, \&)$-задача, що полягає у відповіді на питання «Чи існує така розмітка, при якій всі мітки та пари міток є допустимими?».

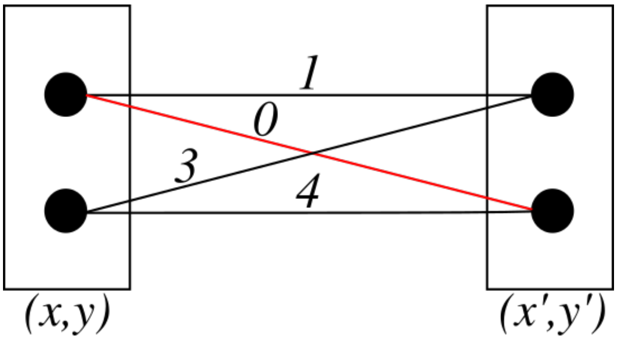
На початку всі вершини вважаються допустимими, тобто $\mu_{(x,y)}(d) = 1$ для всіх міток $d \in D$ в кожному об'єкті $(y,x) \in T$. В кожній парі сусідніх об'єктів $((x, y), (x', y')) \in \mathcal{N}$ знаходиться дужка з мінімальною вагою $\min\limits_{d,d' \in D} g_{(x,y),(x',y')}(d, d') = g_{(x,y),(x',y')}\left(\tilde{d}, \tilde{d'}\right)$. Між цією парою об'єктів допустимими є ті дужки, вага яких відрізняється від мінімальної не більше наперед заданої малої величини $\varepsilon \in \mathbb{R}_+$ (рис. 2.6), тобто $\mu_{(x,y),(x',y')}(d, d') = 1$, якщо

$$g_{(x,y),(x',y')}(d, d') - g_{(x,y),(x',y')}\left(\tilde{d}, \tilde{d'}\right) \le \varepsilon.$$

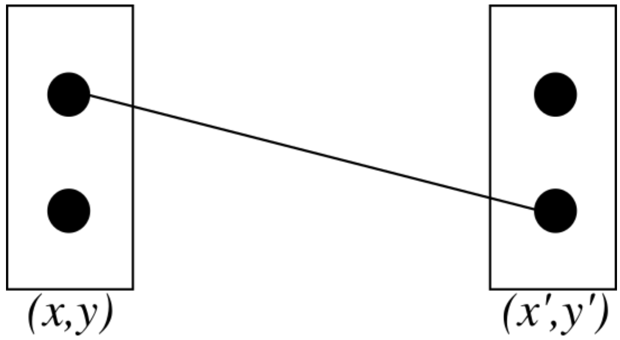
Якщо ця нерівність не виконується, дужка вважається не допустимою, відповідна змінна $\mu_{(x,y),(x',y')}(d, d') = 0$.

Алгоритм полягає в багатократному застосуванні *операції «викреслювання вершини»* (рис. 2.7)

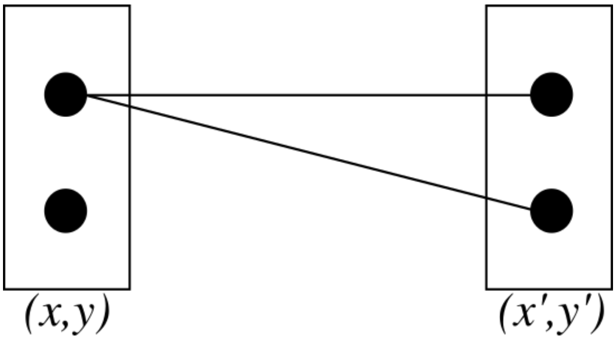
$$\mu_{(x,y)}(d) = \mu_{(x,y)}(d) \,\&\, \mathop{\&}_{(x',y') \in \mathcal{N}(x,y)} \bigvee_{d' \in D} \mu_{(x,y),(x',y')}(d, d')$$



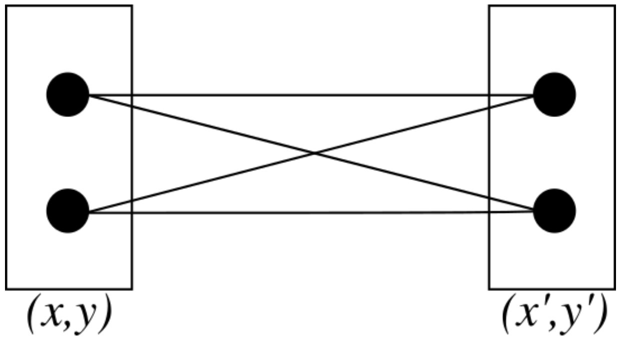
(а) Штрафи за вибір пари міток між двома сусідніми об'єктами. Дужка з мінімальним штрафом позначена червоним



(б) Допустимі дужки при $\varepsilon = 0.5$



(в) Допустимі дужки при $\varepsilon = 1$



(г) Допустимі дужки при $\varepsilon = 4$

Рисунок 2.6 — Допустимі дужки при різних значеннях ε

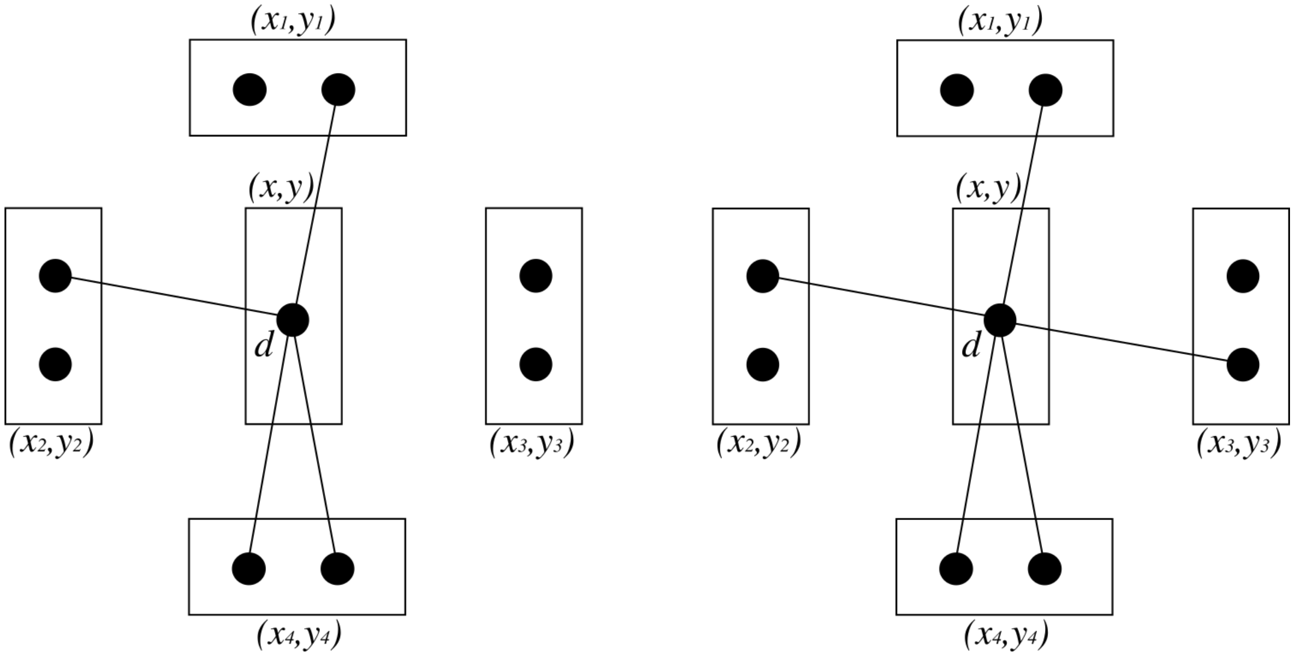
та *операції «викреслювання дужки»* (рис. 2.8)

$$\mu_{(x,y),(x',y')}\left(d, d'\right) = \mu_{(x,y),(x',y')}\left(d, d'\right) \,\&\, \mu_{(x,y)}\left(d\right) \,\&\, \mu_{(x',y')}\left(d'\right).$$

Алгоритм завершує роботу зі скінченну кількість ітерацій, адже ніяка викреслена вершина або дужка не може знов стати допустимою. Якщо після завершення роботи алгоритму деякі вершини залишились допустимими, то було зроблено досить ітерацій дифузії, і з множини допустимих вершин можна побудувати розмітку.

Таким чином, для знаходження найкращої розмітки, необхідно

1) виконати N ітерацій алгоритму дифузії, де N — фіксоване число;
2) запустити алгоритм викреслювання другого порядку;



(а) З вершини (x, y, d) до об'єкту (x, y) всі дужки викреслені (не допустимі), тому вершина (x, y, d) теж стає недопустимою

(б) Вершину (x, y, d) поєднує допустима дужка хоча б з однією вершиною в кожному сусідньому об'єкті. Вершина (x, y, d) залишається допустимою

Рисунок 2.7 — Операція «викреслювання вершини». Зображені лише допустимі дужки

3) якщо після зупинки алгоритму викреслювання залишились допустимі вершини, вибрати найкращу розмітку, інакше — перейти до першого кроку.

Для гарантії існування розмітки після застосування алгоритму дифузії штрафи за вибір пари міток g вихідного графу повинні мати властивість субмодулярності [8].

Для введення субмодулярності необхідно задати порядок на множині міток D. В задачі стереобачення, яка розглядається, мітки задаються цілими числами, які можна впорядкувати по зростанню або спаданню. Будемо вважати, що серед двох міток $d, d' \in D$ *вищою* є та мітка, значення якої менше, як зображено на рисунку 2.2. Тобто якщо $d < d'$, то d є вищою за d'.

Означення. Функція g є *субмодулярною*, якщо для будь-якої пари сусі-

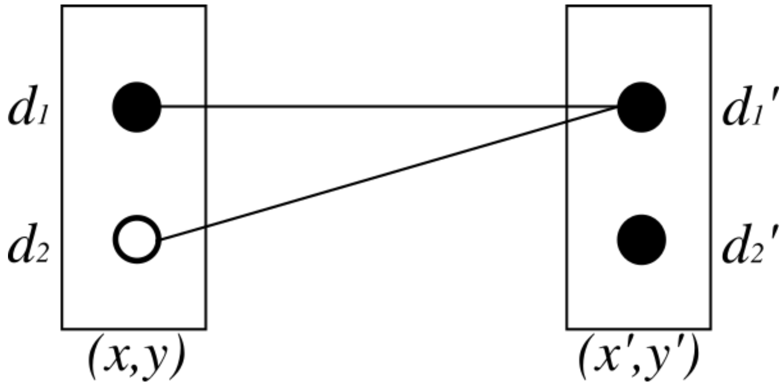


Рисунок 2.8 — Операція «викреслювання дужки». Зафарбованими кружечками позначені допустимі вершини, незафарбованими — недопустимі вершини. Дужка, що поєднує мітку d_2 в об'єкті (x, y) та мітку d_1' в об'єкті (x', y'), буде викреслена, бо одна з цих міток, а саме d_2, є недопустимою. Дужка, що поєднує мітку d_1 в об'єкті (x, y) та мітку d_1' в об'єкті (x', y'), залишиться допустимою

дніх об'єктів $((x, y), (x', y')) \in \mathcal{N}$ та будь-яких пар міток $d_1 > d_2$ та $d_1' > d_2'$ (рис. 2.9) виконується нерівність

$$\begin{gathered} g_{(x,y),(x',y')}(d_1, d_1') + g_{(x,y),(x',y')}(d_2, d_2') \leq \\ \leq g_{(x,y),(x',y')}(d_1, d_2') + g_{(x,y),(x',y')}(d_2, d_1') \,. \end{gathered} \tag{2.20}$$

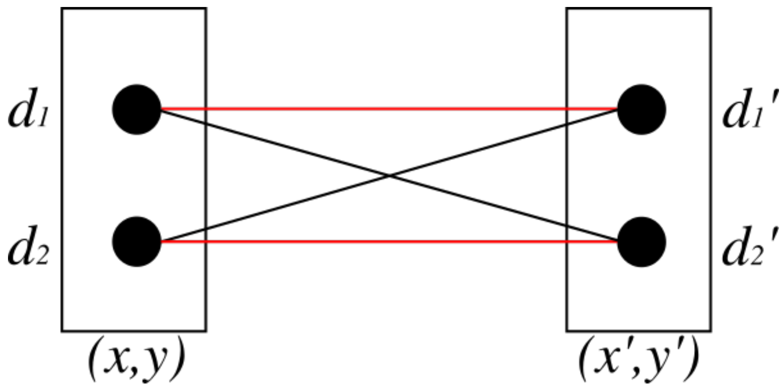


Рисунок 2.9 — Для субмодулярної функції сума паралельних дужок (позначені червоним) не перевищує суми дужок, що перетинаються (позначені чорним)

Прикладом штрафної функції для дужок з властивістю субмодулярності є модуль різниці між значеннями міток

$$g_{(x,y),(x',y')}\left(d, d'\right) = \left|d - d'\right|. \tag{2.21}$$

Для доведення цього твердження достатньо розглянути два сусідніх об'єкта $((x, y), (x', y')) \in \mathcal{N}$ з двома мітками в кожному: $d_1 > d_2$ та $d_1' > d_2'$ та розглянути всі можливі варіанти значень міток (рис. 2.10). Для всіх можливих випадків, що зображені на рисунку, необхідно розкрити модулі в нерівності (2.20) та переконатися, що вони виконуються.

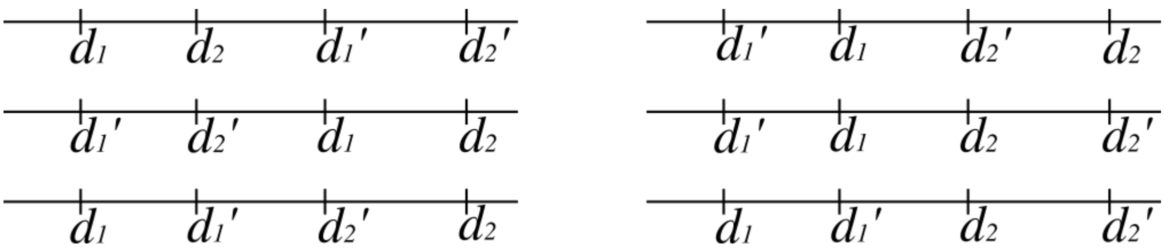


Рисунок 2.10 — Ілюстрація для доведення того факту, що (2.21) — субмодулярна функція

Якщо після того, як алгоритм викреслювання зупинився, в кожному об'єкті $(x, y) \in T$ залишилась хоча б одна допустима вершина, то в кожному об'єкті обирається та мітка, яка є найвищою. Сукупність таких міток і буде утворювати оптимальну розмітку.

Висновки до розділу 2

Пред'явлено розв'язок задачі стереобачення за допомогою алгоритма дифузії. Доведено, що кожний крок алгоритму дифузії не зменшує дуальну функцію Лагранжа для енергії задачі, а також наведені інші важливі властивості. Описано, як знайти оптимальну розмітку після розв'язання задачі оптимізації.

3 СЕГМЕНТАЦІЯ ЗОБРАЖЕННЯ ДЛЯ ПРИСКОРЕННЯ АЛГОРИТМІВ СТЕРЕОБАЧЕННЯ

У третьому розділі запропоновано спосіб прискорення алгоритмів стереобачення на прикладі алгоритму дифузії за допомогою сегментації зображення.

3.1 Сегментація зображення

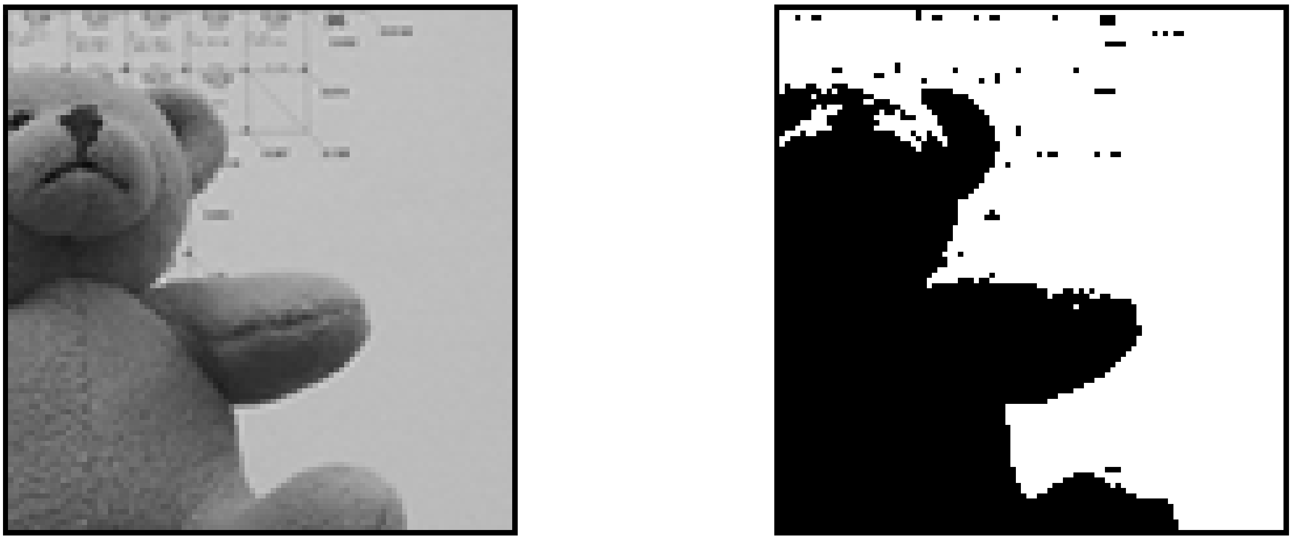
Ліве зображення L з вхідної стереопари розбивається на прямокутну решітку з однаковим заданим розміром комірок. Всі пікселі, що належать одній комірці, діляться на дві групи за середньою інтенсивністю пікселів комірки (рис. 3.1). До першої групи (зображена білим на рисунку) відносять пікселі, інтенсивність яких не перевищує середньої інтенсивності пікселів комірки. Піксель з координатами (x^*, y^*), що належить даній комірці S потрапляє до першої групи, якщо

$$L\left(x^*, y^*\right) \leq \frac{1}{|S|} \sum_{(x,y)\in S} L\left(x, y\right),$$

де $|S|$ – кількість пікселів у комірці. До другої групи, що зображена чорним на рисунку, відносять всі інші пікселі комірки, що не потрапили в першу групу. Кожну таку групу будемо називати *суперпікселем*. Таким чином, в процесі сегментації ліве зображення стереопари розбивається на комірки, в яких знаходиться два суперпікселя: більш світлий і більш темний.

Після запропонованої сегментації будується $|T_s|$-дольний граф з множиною координат суперпікселів

$$T_s = \left\{(x_s, y_s, i) \mid 1 \leq x_s \leq m,\, 1 \leq y_s \leq n,\, i \in \{0, 1\}\right\},$$



(а) Комірка зображення

(б) Комірка зображення, розбита на два суперпікселя

Рисунок 3.1 — Візуалізація суперпікселів

де m — кількість комірок по горизонталі, n — кількість комірок по вертикалі. Тепер кожній долі графу відповідає один суперпіксель — об'єкт графу. Кожен об'єкт $(x_s, y_s, i) \in T_s$ містить $|D|$ міток (або вершин), які відповідають всім можливим зсувам горизонтальної координати пікселів, що належать об'єкту. Таким чином, всі пікселі, що належать одному суперпікселю, будуть мати однакову глибину (або паралакс) на результуючій карті глибин, тобто тепер горизонтальний зсув $d \in D$ буде шукатись не для кожного пікселя окремо, а для цілої групи пікселів, які утворюють суперпіксель, одразу.

В об'єкті $(x_s, y_s, i) \in T_s$ на вершину з міткою $d \in D$ накладається штраф, який дорівнює сумі штрафів за вибір відповідних вершин у всіх пікселях, які належать даному суперпікселю,

$$f^s_{(x_s,y_s,i)}\left(d\right) = \sum_{(x,y)\in(x_s,y_s,i)} f_{(x,y)}\left(d\right),$$

де $f_{(x,y)}\left(d\right)$ — штраф за вибір вершини, введений в першому розділі в формулі (1.4).

Кожен об'єкт може мати до дев'яти сусідів: по два об'єкти в верхній, правій, нижній і лівій комірках, а також другий об'єкт, який належить тій же комірці (рис. 3.2). Об'єкти, що відповідають коміркам на краях зображення, мають по сім сусідів, а об'єкти, що відповідають кутовим коміркам, — по п'ять. Штраф, який накладається на дужки між вершинами різних об'єктів, такий же, як і при постановці задачі без суперпікселів, тобто дорівнює $g_{(x_s,y_s,i),(x'_s,y'_s,i')}(d,d')$, де $(x_s,y_s,i) \in T_s$ та $(x'_s,y'_s,i') \in T_s$ — сусідні об'єкти, d — мітка в об'єкті (x_s,y_s,i), а d' — мітка в об'єкті (x'_s,y'_s,i'). Множину всіх пар сусідніх об'єктів в цьому графі позначимо через $\mathcal{N}^s$.

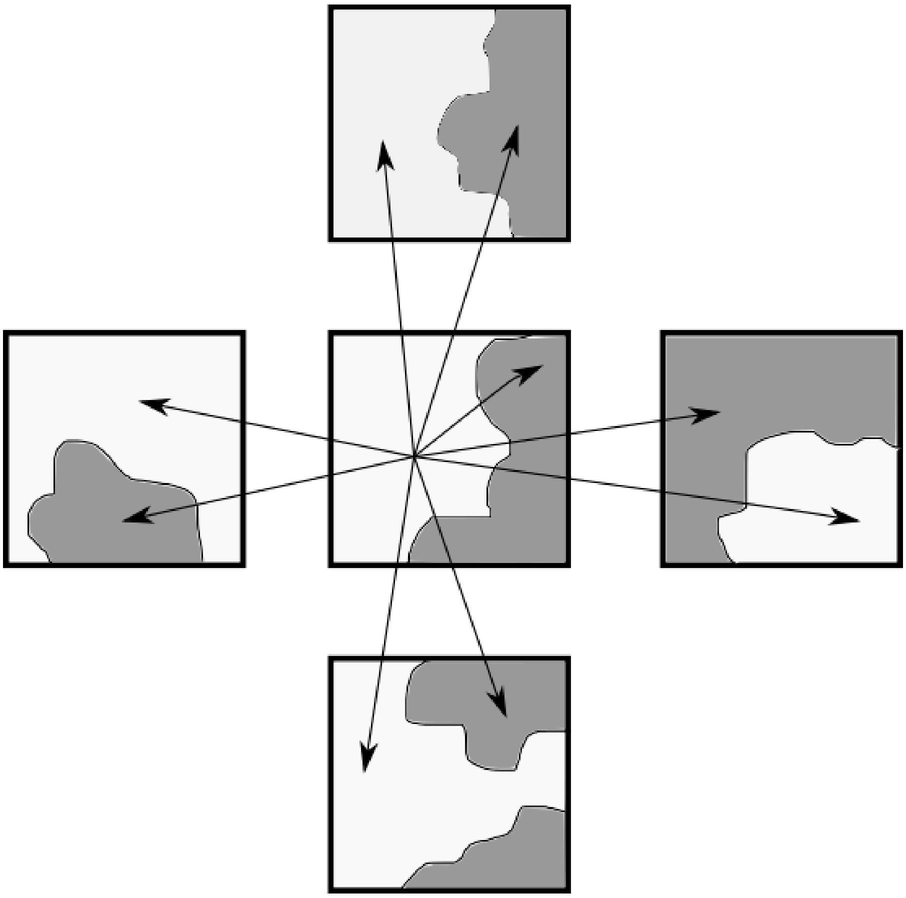


Рисунок 3.2 — Структура сусідства при використанні суперпікселів. Квадратами позначені комірки, що містять по два суперпікселя (об'єкта): світлий ($i = 0$) і темний ($i = 1$). Стрілками позначені дуги, що виходять зі світлого суперпікселя центральної комірки в об'єкти, що є для нього сусідніми

Аналогічно штрафній функції (1.5) вихідної задачі отримуємо штрафну функцію модифікованої задачі

$$G_s(\boldsymbol{d}) = \sum_{x=1}^{m} \sum_{y=1}^{n} \sum_{i \in \{0,1\}} f^s_{(x_s,y_s,i)}\left(d\left(x_s, y_s, i\right)\right) +$$
$$+ \sum_{((x_s,y_s,i),(x'_s,y'_s,i')) \in \mathcal{N}^s} g_{(x_s,y_s,i),(x'_s,y'_s,i')}\left(d\left(x_s, y_s, i\right), d\left(x'_s, y'_s, i'\right)\right).$$

Далі задача розв'язується методом, що описаний у попередньому розділі.

3.2 Складність алгоритму дифузії

Висновки до розділу 3

Запропоновано спосіб прискорення розв'язання задачі стереобачення за допомогою алгоритма дифузії. Прискорення алгоритму базується на зменшенні розміру графу, на якому розв'язується задача, за допомогою сегментації зображення так, щоб не втратити багато інформації та не сильно погіршити результуючу карту глибин.

4 ПРАКТИЧНІ РЕЗУЛЬТАТИ

У четвертому розділі представлено практичні результати застосування алгоритму дифузії з сегментацією та без із зазначенням вхідних даних.

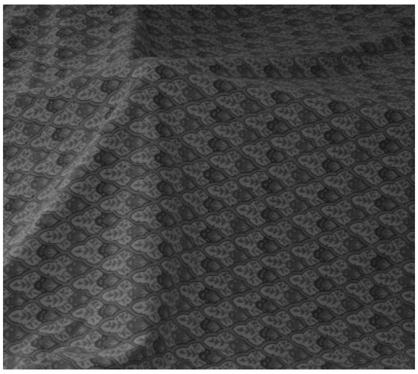
4.1 Практичні результати

Вихідні зображення для побудови карти глибин за допомогою описаних алгоритмів були взяті з набору стереопар, зроблених в Мідлберійському коледжі в 2001 [9], 2003 [10] та 2006 [11] [12] роках. На рисунку 4.1 наведені ліві та праві зображення зі стереопар, для яких будувались карти глибин в даному розділі дисертації.

На рисунку 4.2 зображені карти глибин, отримані за допомогою алгоритму дифузії, де кожній долі графу відповідає один піксель, як описано в другому розділі дисертації. Для кожного зображення вказано кількість ітерацій, зроблений алгоритмом дифузії, час виконання всіх ітерацій, а також кількість міток $|D|$ в графі.

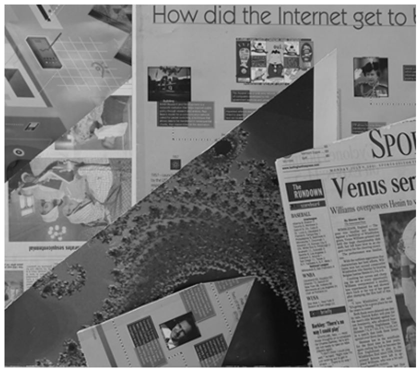
На рисунку 4.3 зображені карти глибин, отримані за допомогою алгоритму дифузії з сегментацією зображення, що описана в попередньому розділі дисертації, з розміром комірок 5 на 5 пікселів. Для кожного зображення вказано кількість ітерацій, зроблений алгоритмом дифузії, час виконання всіх ітерацій, а також кількість міток $|D|$ в графі. Отримані досить гладкі карти глибин, однак видніються неточності на краях об’єктів.

Для порівняння на рисунку 4.4 наведені карти глибин, отримані з тими самими штрафними функціями та значеннями параметрів за допомогою динамічного програмування, де для кожного рядка зображення карта глибин шукається незалежно. Хоча алгоритм і працює дуже швидко, на отриманих картах глибин чітко видно горизонтальні лінії, які є результатом того, що всі рядки

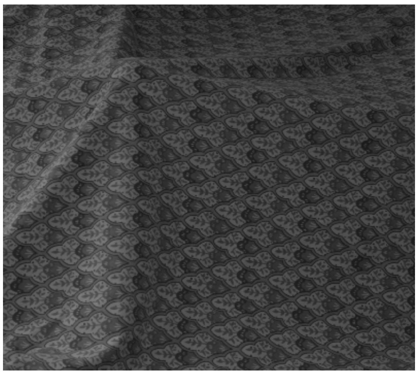


(а) Тканина (Cloth1), 400×355 пікселів

(б) Квіткові горщики (Flowerpots), 400×338 пікселів

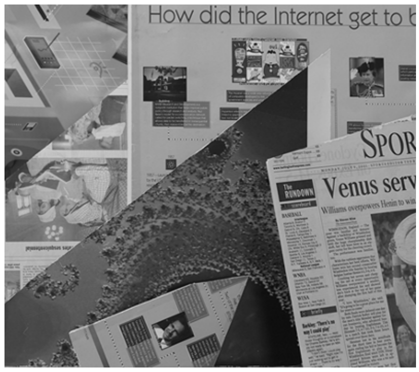


(в) Плакат (Poster), 400×352 пікселяв



(г) Тканина (Cloth1), 400×355 пікселів

(д) Квіткові горщики (Flowerpots), 400×338 пікселів



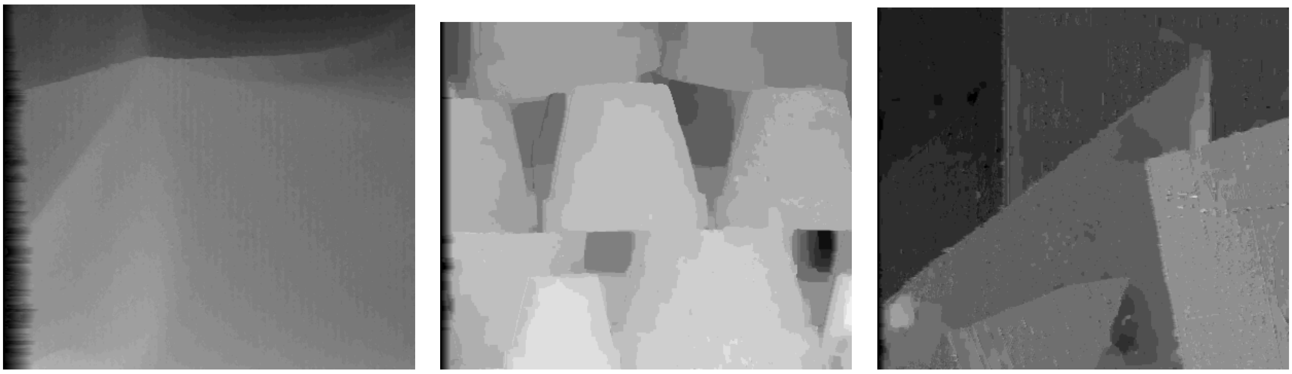
(е) Плакат (Poster), 400×352 пікселів

Рисунок 4.1 — Зображення стереопар, для яких будувались карти глибин в цьому розділі. Перший рядок містить ліві зображення, другий — праві

зображення обробляються незалежно. Таким чином, не алгоритм не враховує гладкість об’єктів. Даний метод описано в першому розділі дисертації.

На рисунку 4.5 наведені істинні синтезовані карти глибин, надані разом за набором стереопар.

Карти глибин, отримані алгоритмом дифузії не є такими гладкими, як ідеальні карти глибин. Для підвищення гладкості карт глибин та позбавлення від артефактів після застосування алгоритмів стереобачення часто використовують згладжуючі фільтри та інші методи обробки карти глибин. Приклад



(а) $2'400$ ітерацій, 5 годин 40 хвилин, $|D| = 40$

(б) $2'800$ ітерацій, 1 година 20 хвилин, $|D| = 16$

(в) $1'600$ ітерацій, 46 хвилин, $|D| = 16$

Рисунок 4.2 — Карти глибин, отримані алгоритмом дифузії без сегментації зображення

одного з методів покращення карти глибин [13] наведений на рисунку 4.6. В даному методі для кожного пікселя розглядається віконце певного розміру з центром в цьому пікселі. Значення зсуву для даного пікселя перераховується шляхом голосування з використанням сусідніх пікселів. Далі використовується медіанний фільтр для згладжування зображення. Він прибирає імпульсний шум, позбавляючи карту глибин від деяких артефактів.

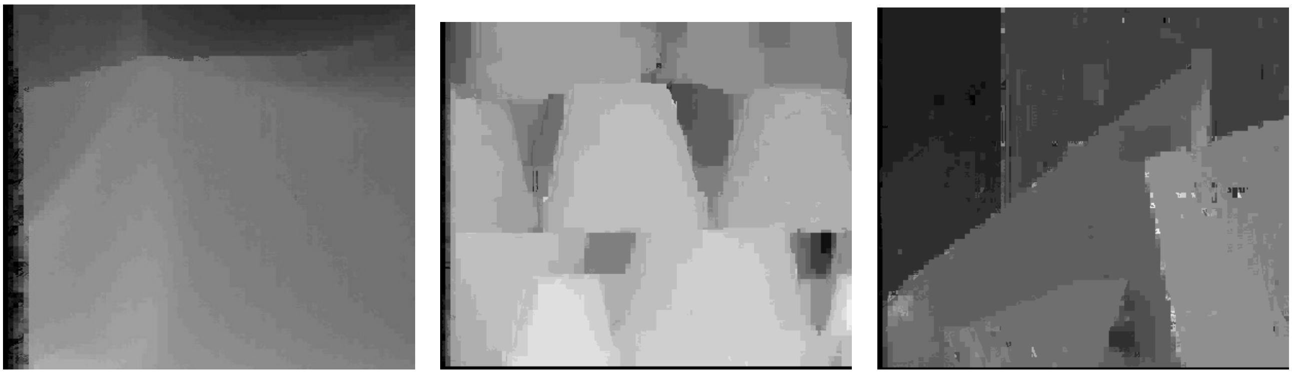
В якості штрафної функції для вершин f в даній роботі використовувався модуль різниці інтенсивностей відповідних пікселів на двох зображеннях

$$f_{(x,y)}\left(d\right) = \left|L\left(x,y\right) - R\left(x-d,y\right)\right|,$$

$(x,y) \in T$, $d \in D$, а в якості штрафної функції для дужок g — модуль різниці вибраних зсувів $d \in D$ та $d' \in D$ в сусідніх об'єктах графу

$$g_{(x,y),(x',y')}\left(d,d'\right) = \alpha \cdot \left|d - d'\right|,$$

де $\left(\left(x,y\right),\left(x',y'\right)\right) \in \mathcal{N}$, а $\alpha = 1.4$ — коефіцієнт згладжування, підібраний експериментальним шляхом. В попередньому розділі вказано, що $|d-d'|$ —

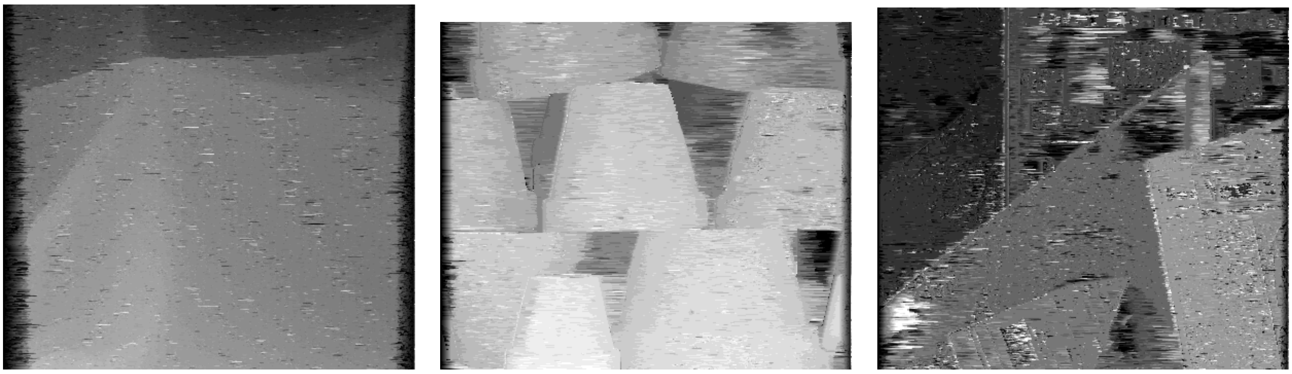


(а) 100 ітерацій, 19 хвилин, $|D| = 40$

(б) 450 ітерацій, 15 хвилин, $|D| = 16$

(в) 400 ітерацій, 14 хвилин, $|D| = 16$

Рисунок 4.3 — Карти глибин, отримані алгоритмом дифузії після сегментації зображення



(а) 1.825 секунди

(б) 0.792 секунди

(в) 0.83 секунди

Рисунок 4.4 — Карти глибин, отримані за допомогою динамічного програмування

субмодулярна функція від величин $d, d' \in D$. Після домноження на константу $\alpha \in \mathbb{R}$ функція залишається субмодулярною, що гарантує існування розмітки після розв'язання оптимізаційної задачі алгоритмом дифузії.

Додатково були введені обмеження на можливі мітки в кожному об'єкті: для об'єкта $(x, y) \in T$ з горизонтальною координатою x не може бути обраний зсув $d \in D$, такий що $d > x$, який би перевів горизонтальну координату пікселя у від'ємне число $x - d < 0$ (рис. 4.7). Дані обмеження були введені за

(а) Тканина (Cloth1) (б) Квіткові горщики (Flowerpots) (в) Плакат (Poster)

Рисунок 4.5 — Істинні карти глибин

допомогою штрафної функції f

$$f_{(x,y)}(d) = \begin{cases} |L(x,y) - R(x-d,y)|, & x \geq d, \\ +\infty, & x < d. \end{cases}$$

Також були накладені обмеження на зсуви в сусідніх об’єктах по горизонталі: $d' \leq d + 1$, де $d \in D$ — мітка в об’єкті $(x, y) \in T$, d' — мітка в об’єкті $(x', y') \in \mathcal{N}(x, y)$, такому що $x' = x + 1$.

Алгоритм був реалізований на мові програмування Rust. В ході роботи був використаний комп’ютер з процесором Intel(R) Core(TM) i5-7400 і ОЗП DDR4 2133MHz.

Висновки до розділу 4

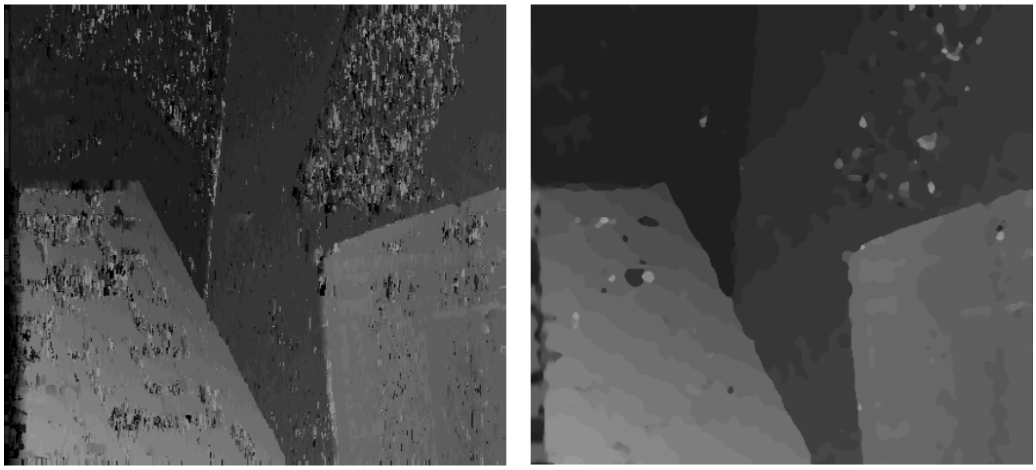


Рисунок 4.6 — Приклад роботи одного з методів покращення карти глибин. Зліва зображена початкова карта глибин, отримана алгоритмом стереобачення, справа — покращена карта глибин. Зображення взяті зі статті 4.6

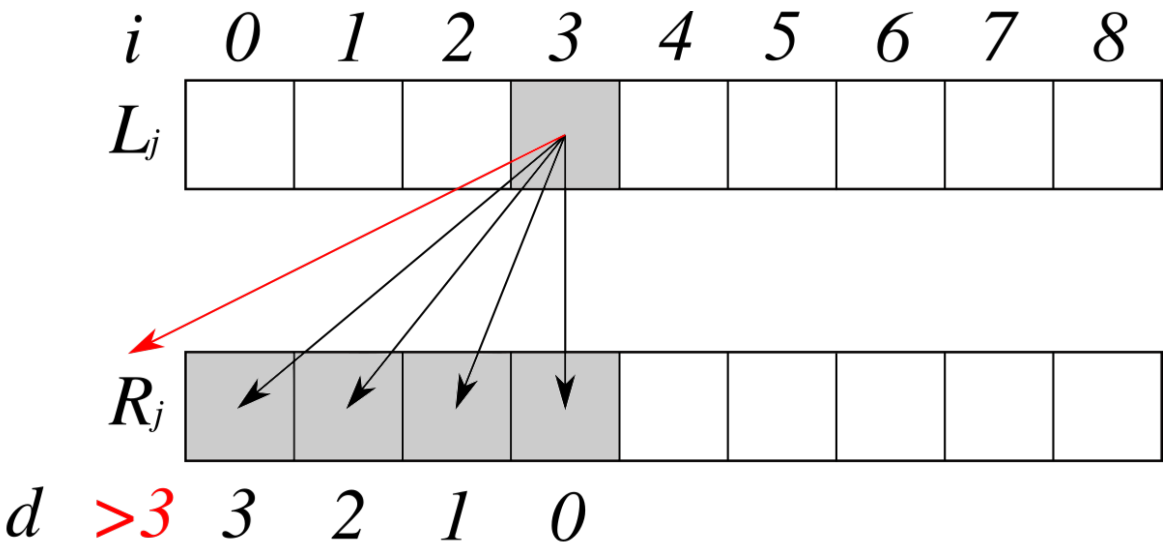


Рисунок 4.7 — Обмеження на вибір мітки в об'єкті. L_j та R_j — рядок під номером j лівого та правого зображення відповідно. Допустимі зсуви $d \in \{0, 1, 2, 3\}$ для даного об'єкта (i, j) з горизонтальною координатою $i = 3$ зображено чорними стрілками, недопустимі зсуви $d > 3$ — червоною стрілкою

ВИСНОВКИ

Побудова карти глибин — важка задача та на сьогоднішній день не розв’язана точно. Обчислювальна складність деяких алгоритмів розв’язання задачі стереобачення досить велика, що призводить до великої тривалості їх роботи.

Дана робота містить постановку задачі стереобачення, а також її розв’язання алгоритмом дифузії. Описано новий спосіб прискорення алгоритму за допомогою сегментації зображення, при якому не втрачається багато інформації про глибину об’єктів. При незначних втратах якості отриманої карти глибин вдалося прискорити роботу алгоритму в декілька разів.

ПЕРЕЛІК ПОСИЛАНЬ

1 A Maximum Likelihood Stereo Algorithm / Ingemar J. Cox, Sunita L. Hingorani, Satish B. Rao, Bruce M. Maggs // *Comput. Vis. Image Underst.* — 1996. — May. — Vol. 63, no. 3. — Pp. 542–567. https://doi.org/10.1006/cviu.1996.0040.

2 Wang, Fuzhi. Stereo Matching Using Iterative Dynamic Programming Based on Color Segmentation of Images / Fuzhi Wang, Changlin Song, Qiang Du // *Journal of Computers*. — 2014. — 06. — Vol. 9.

3 Geiger, Andreas. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite / Andreas Geiger, Philip Lenz, Raquel Urtasun // Conference on Computer Vision and Pattern Recognition (CVPR). — 2012.

4 Luo, Wenjie. Efficient Deep Learning for Stereo Matching / Wenjie Luo, Alexander G. Schwing, Raquel Urtasun // *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. — 2016. — Pp. 5695–5703.

5 Kolmogorov, Vladimir. Graph Cut Algorithms for Binocular Stereo with Occlusions / Vladimir Kolmogorov, Ramin Zabih // Handbook of Mathematical Models in Computer Vision. — 2005.

6 Feng, Liting. Superpixel-based graph cuts for accurate stereo matching / Liting Feng, Kaihuai Qin // *IOP Conference Series: Earth and Environmental Science*. — 2017. — 06. — Vol. 69. — P. 012161.

7 Savchynskyy, Bogdan. Discrete Graphical Models — An Optimization Perspective / Bogdan Savchynskyy // *Foundations and Trends® in Computer Graphics and Vision*. — 2019. — Vol. 11, no. 3-4. — Pp. 160–429. http://dx.doi.org/10.1561/0600000084.

8 Шлезингер, М. И. Анализ алгоритмов диффузии для решения оптимизационных задач структурного распознавания / М. И. Шлезингер, К. В. Антонюк // *Кибернетика и системный анализ*. — 2011.

9 Scharstein, D. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms / D. Scharstein, R. Szeliski, R. Zabih // Proceedings IEEE Workshop on Stereo and Multi-Baseline Vision (SMBV 2001). — 2001. — Pp. 131–140.

10 Scharstein, D. High-accuracy stereo depth maps using structured light / D. Scharstein, R. Szeliski // 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings. — Vol. 1. — 2003. — Pp. 195–202.

11 Scharstein, Daniel. Learning Conditional Random Fields for Stereo / Daniel Scharstein, Chris Pal. — 2007. — 06.

12 Hirschmüller, Heiko. Evaluation of Cost Functions for Stereo Matching / Heiko Hirschmüller, Daniel Scharstein. — 2007. — 06.

13 Touzene, Nadia. Accurate Real-Time Disparity Map Computation Based on Variable Support Window / Nadia Touzene, Slimane Larabi // *International Journal of Artificial Intelligence & Applications*. — 2011. — 07. — Vol. 2. — Pp. 22–34.