

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

КАФЕДРА КОМП'ЮТЕРНИХ НАУК

Секція інформаційно-комунікаційних технологій

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

на тему:

**«Інтелектуальна технологія розпізнавання цілісності
та просадки стічних труб»**

Завідувач

випускаючої кафедри

Довбиш А.С.

Керівник роботи

Коробов А.Г.

Студент групи ІК.мз – 01с

Шикула М.М.

СУМИ 2021

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

Кафедра комп'ютерних наук

Затверджую _____

Зав. кафедрою Довбиш А.С.

“ _____ ” _____ 2021 г.

ЗАВДАННЯ

до випускної роботи

Студента другого курсу мігістратури, групи ІК.мз-01с спеціальності “Інформаційно-комунікаційні технології” заочної форми навчання Шикули Миколи Миколайовича.

Тема: “Інтелектуальна технологія розпізнавання цілісності та просадки стічних труб ”

Затверджена наказом по СумДУ

№ _____ от _____ 2021 г.

Зміст пояснювальної записки: 1) аналіз проблеми та постановка задачі; 2) інформаційна технологія аналізу відеоданих інспекції стічних труб; 3) створення набору даних для навчання двох моделей 4) навчання та оцінка двох згорткових нейромереж 5) реалізація алгоритму інтелектуальної технології розпізнавання цілісності та просадки стічних труб;

Дата видачі завдання “ _____ ” _____ 2021 г.

Керівник магістерської роботи _____ Коробов А.Г.

Завдання прийняв до виконання _____ Шикула М.М.

РЕФЕРАТ

Записка: 60 стор., 29 рис., 2 табл., 1 додаток, 46 джерело.

Об'єкт дослідження — застосування згорткових нейронних мереж для інспекції стічних труб.

Мета роботи — розробка алгоритму обробки відеоданих інспекції стічних труб, з використанням глибоких нейронних мереж, для детектування функціональних дефектів і рівня води в трубах та формуванням звіту з рівнем просадки стічних труб

Методи дослідження — аналіз відеоданих інспекції, синтез інтелектуальної системи

Результати — розроблено інтелектуальну систему, що визначає рівень просадки стічної труби за рівнем води, відносно даних лічильника пройденого камерою шляху на відеоданих інспекції. Для цього використовується дві моделі штучних згорткових нейронних мереж: модель нейромережі для розпізнавання цілісності стічних труб та модель нейромережі для розпізнавання рівня води в стічній трубі, що працюють по черзі. В результаті створюються звіти в форматі таблиці з мітками відстані, часу та рівнем просадки.

ГЛИБОКІ НЕЙРОННІ МЕРЕЖІ, СТІЧНІ ТРУБИ,
ДЕТЕКТУВАННЯ РІВНЯ ПРОСАДКИ ТРУБ ЗА РІВНЕМ ВОДИ,
ВІДЕОІНСПЕКЦІЯ,
ІНТЕЛЕКТУАЛЬНА СИСТЕМА

ЗМІСТ

ВСТУП.....	5
1. АНАЛІЗ АКТУАЛЬНОСТІ ПРОБЛЕМИ І ПОСТАНОВКА ЗАДАЧІ.....	7
1.1 Сучасні підходи в інспектуванні каналізаційної мережі.....	10
1.2 Аналіз методів та моделей машинного навчання і комп'ютерного зору...	24
1.3 Постановка задачі.....	33
2. МОДЕЛІ ТА МЕТОДИ АНАЛІЗУ ВІДЕОІНСПЕКЦІЇ СТІЧНИХ ТРУБ.....	34
2.1 Модель нейромережі для розпізнавання цілісності стічних труб.....	34
2.2 Модель нейромережі для розпізнавання рівня води в стічній трубі.....	36
2.3 Методи навчання системи розпізнавання.....	37
2.4 Критерії оцінки якості роботи системи.....	38
3. РЕАЛІЗАЦІЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ РОЗПІЗНАВАННЯ ЦІЛІСНОСТІ ТА ПРОСАДКИ СТІЧНИХ ТРУБ.....	40
3.1 Формування навчальних та тестових даних.....	40
3.2 Результати машинного навчання нейромережі.....	41
3.3 Опис алгоритму роботи інтелектуальної системи.....	44
3.4 Програмна реалізація.....	44

ВИСНОВКИ.....	45
СПИСОК ЛІТЕРАТУРИ.....	46
Додаток А.....	50

ВСТУП

Життєдіяльність людського суспільства неможлива без споживання води, крім того, вода є основним та головним ознакою існування всього живого. Вода, подається споживачеві системою водопостачання, в процесі використання в основному втрачає початкову якість, перетворюючись на стічної води, і підлягає видаленню із зони споживання. Цей процес однаковий як при використанні води населенням як на господарсько-побутові потреби, так і в результаті здійснення технологічних процесів на виробничих підприємствах різного призначення. До стічних вод відносять також дощові та талі води. Стічні води будь-якого походження отримують додаткові хімічні та механічні домішки (забруднення), що змінюють початковий хімічний склад та фізичні властивості води, що потребує їх організованого збору, транспортування, очищення і повернення в навколишнє середовище (як правило природні водойми). Для цього необхідний комплекс інженерних пристроїв та обладнання, що становлять систему водовідведення[6].

Одна з головних вимог до системи водовідведення – герметичність трубопроводів та надійність всіх її елементів. Основними проблемами на

мережах каналізації є засмічення та велика зношеність трубопроводів, що виникають внаслідок експлуатації. З досвіду експлуатації водовідвідних мереж одним з діючих та продуктивних методів спостереження є інспекція за допомогою систем відеоспостереження (ССТV), які дозволяють досить точно встановити рівень технічного стану та характер пошкоджень, якщо вони є. З отриманих даних виробляються заходи, які дозволяють максимально ефективно вирішити експлуатаційні завдання[7].

Протяжність каналізаційних труб сягає в Україні десятки тисяч кілометрів, які більш ніж, як на тридцять процентів в аварійному стані. Тому при майбутніх обстеженнях буде збиратися велика кількість відеоматеріалів ССТV, оцінка яких займе багато робочого часу великих команд професіоналів.

Головне завдання цієї магістерської роботи включає в себе опис інтелектуальної технології розпізнавання цілісності та просадки стічних труб за допомогою створення двох моделей згорткових нейронних мереж , які працюють на прогнозування певних видів функціональних дефектів та рівня води для визначення просадки цих труб. Результати цієї роботи допоможуть підвищити рівень висновків по функціональному становищу каналізацій-ного трубопроводу та вплинути на якість звітів відеоспостережень.

1. АНАЛІЗ АКТУАЛЬНОСТІ ПРОБЛЕМИ І ПОСТАНОВКА ЗАДАЧІ

Розподільна система інженерних мереж каналізаційного господарства є складним комплексом споруд для стійкого відведення стічних вод. Ключовими її елементами є трубопроводи водовідведення та каналізаційні тунелі різного діаметра, що характеризують ступінь розвитку та благоустрою міста. В існуючих умовах експлуатуючі підприємства України здійснюють свою діяльність на межі технічних та організаційних можливостей, про що свідчить висока зношеність основних фондів, аварійний стан значної частини мереж за умов недостатнього фінансування галузі [2].

Сумарна протяжність каналізаційних мереж (без Донецької та Луганської областей) у 2016 році дорівнювала — 31140,1 км, у т.ч. ветхих та аварійних — 11438,9 км або 36,7%; протягом 2016 року було замінено 121,2 км або біля 1,1% від потреби. З урахуванням вказаних областей ці показники

складали: всього — 35001,2 км, у т.ч. ветхих та аварійних — 13984,3 км або 40%; замінено 127 км або 0,9% від потреби [18].

В результаті високого рівня зносу, по Україні в рік в середньому трапляється 260 аварій на 100 км трубопроводу[15].

Знос котельних в цілому по Україні складає 55 %, мереж водопровіда – 65 %, мереж каналізації – 63 %, теплових мереж – 63 %, електричних мереж – 58 %, водопровідних насосних станцій – 65 %, каналізаційних насосних станцій – 57 %, очисних споруд водопровіда – 54 %, очисних споруд каналізації – 56 %, трансформаторних підстанцій – 57 %, що свідчить про постійне падіння якості житлово-комунальних послуг[8].

Однією із найбільших проблем підприємств водопровідно-каналізаційного господарства є незадовільний технічний стан їх обладнання, значна зношеність основних засобів (більше 60%), високі питомі витрати енергоресурсів, втрати питної води (35—40%). Ці проблеми є спільними для всієї країни, з причини неефективної експлуатації споруд, невідповідного, застарілого, зношеного обладнання[19].

Нормативно-правова база у сфері водопостачання та водовідведення, в цілому, забезпечує діяльність тільки в сфері централізованого водопостачання та водовідведення. Водночас задекларованою, але недостатньо врегульованою, є сфера нецентралізованого водопостачання та водовідведення. Крім того, профільні нормативно-правові акти сфери водопостачання та водовідведення не в повній мірі відповідають вимогам Європейських директив[19].

Слід зазначити, що оптимальним варіантом розв'язання проблем сфери водопостачання та водовідведення була і є реалізація державної політики щодо розвитку та реконструкції систем централізованого водопостачання та водовідведення; охорони джерел питного водопостачання; доведення якості питної води до вимог державних стандартів; нормативно-правового забезпечення у сфері питного водопостачання та водовідведення; розроблення та впровадження науково-

дослідних і дослідно-конструкторських розробок із застосуванням новітніх матеріалів, технологій, обладнання та приладів, яка відображається в Загальнодержавній програмі "Питна вода України" на 2011—2020 роки" [30].

Очікуваних результатів виконання Програми у 2012—2015 роках не досягнуто. Нормативно-правове та науково-технічне виконання Програми взагалі провалено. Заходи щодо доведення якості питної води до встановлених нормативів викону валися недостатньо і несвоєчасно. Відсутність коштів на обслуговування систем (установок) доочищення унеможлиблює їх експлуатацію, а кошти на їх впровадження використано непродуктивно. Інвентаризація каналізаційних очисних споруд досі не завершена. Оптимізацію роботи систем централізованого водопостачання та водовідведення не здійснено[20].

За даними Рахункової палати України у рамках Програми, яка виконується вже 10 років, за її організаційного та фінансового забезпечення вирішити проблему забезпечення громадян питною водою в необхідних обсягах та відповідно до встановлених нормативів практично нереально. Результати аудиту дають підстави стверджувати, що і до 2020 року Програма виконана не буде. Чинне законодавство не забезпечує цілісного регулювання питань забезпечення громадян питною водою. Заходи з розвитку та реконструкції централізованих систем водопостачання та водовідведення здійснюються безсистемно і непослідовно. Постійного джерела фінансування, насамперед коштів на реконструкцію або будівництво цих систем, практично немає. За відсутності контролю з боку Кабінету Міністрів України та Мінрегіону як державного замовника Програми, оновлена у 2011 році Програма не сприяла кардинальному оновленню підприємств водопровідно-каналізаційного господарства. Суттєвих змін у забезпеченні громадян якісною питною водою протягом 2012—2015 років не відбулося [20].

Говорячи про позитивні приклади інвестування у сфері водопостачання та водовідведення, слід зазначити про два найбільших

проекти, що реалізуються в Україні. Другий проект розвитку міської інфраструктури (англ. Second Urban Infrastructure Project) — найбільший інвестиційний проект Уряду України у сфері водопостачання, водовідведення та поводження з твердими побутовими відходами. Другий найбільший проект у сфері водопостачання та водовідведення реалізується Урядом і передбачає комплексну реконструкцію Бортницької станції аерації міста Києва. Термін реалізації Проекту складає 5 років і триватиме до кінця 2023 року. Кредитні кошти у розмірі 1 млрд дол. США залучаються на пільгових і вигідних для України умовах від Японського агентства міжнародного співробітництва (JICA) [26].

Зазначені проекти є доказом вдалої інвестиційної політики держави в сферу водопостачання та водовідведення[19].

Якщо поглянемо на стан трубопроводів в інших країнах, то виявиться, що більше половини діючих у світі трубопроводів було прокладено ще в другій половині минулого століття, і зараз майже всі вони знаходяться в аварійному стані. Зношеність багатьох із них дуже велика. Тому там використовується більш економічний спосіб заміни труб, такий як, безтраншейний ремонт трубопроводів з широким застосуванням устаткування за допомогою якого здійснюють руйнування старого зношеного трубопроводу та одночасним затягуванням нового[15].

1.1 Сучасні підходи в інспектуванні каналізаційної мережі

Основним завданням технічної експлуатації систем водопостачання та водовідведення є забезпечення надійною роботи всіх елементів[9].

Найвразливіша ланка систем – трубопроводи. Структура та класифікація завдань надійності систем

водопостачання та водовідведення представлена на рис. 1.1

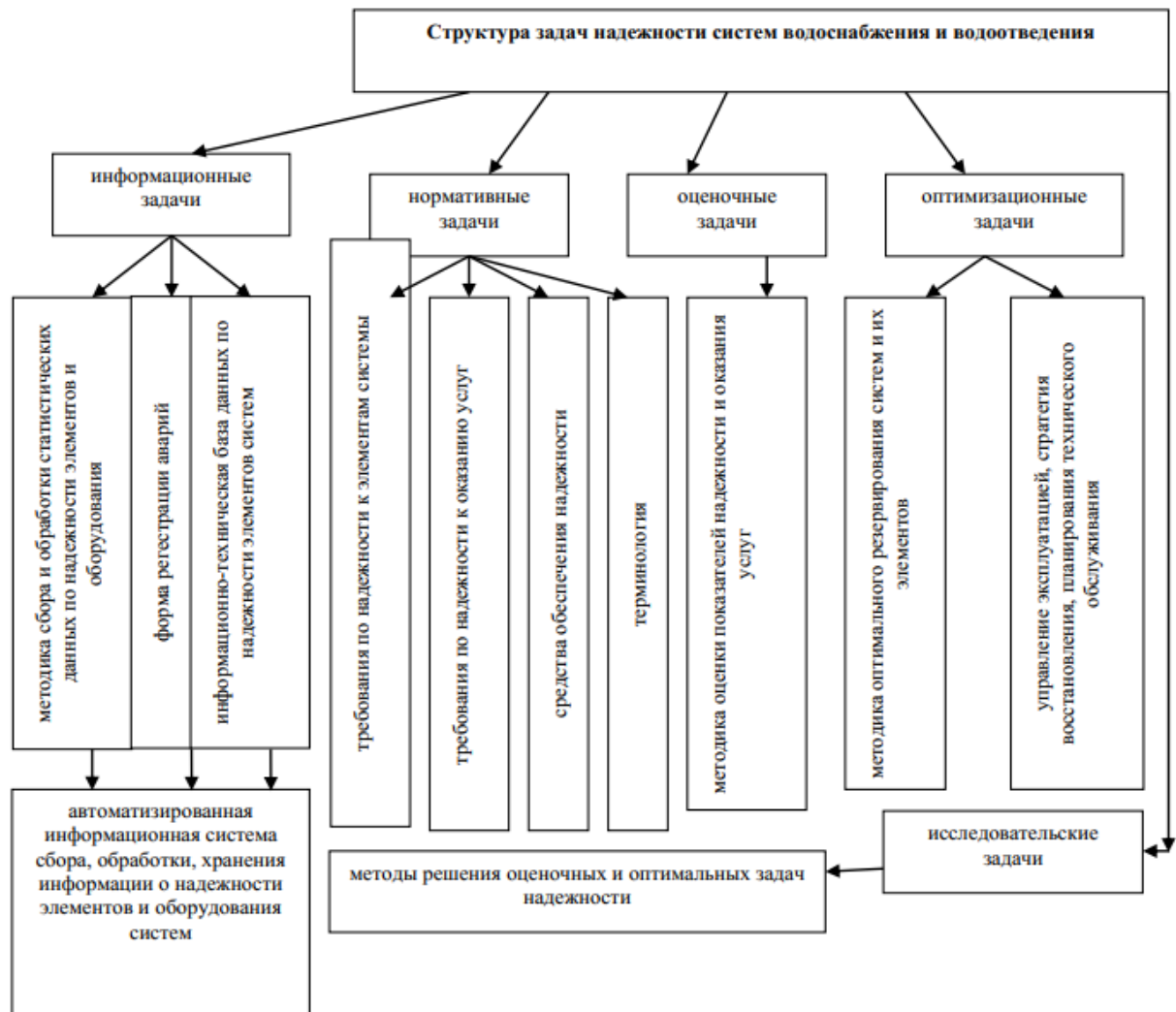


Рисунок 1.1 Структура та класифікація завдань надійності систем

водопостачання та водовідведення

Водопровідна мережа, як і будь-який фізичний об'єкт згодом морально і фізично застаріває. Це потребує проектування нових трубопроводів. Необхідна достовірна вихідна інформація, без якої практично неможливо якісно розробити проектні рішення та виконання будівельних робіт на потенційних об'єктах реновації, а також ефективна експлуатація трубопровідної мережі [14].

Для прийняття правильного рішення щодо розробки стратегії та тактики реновації трубопроводних мереж, різного призначення, проектувальник повинен мати дані щодо діагностичних обстежень старих трубопроводів, використовувати їх у своїй роботі і чітко уявити картину можливою реабілітації окремих ділянок трубопроводної мережі з використанням сучасних методів будівництва [14,16,17].

На інтенсивність відмови мереж водопостачання та водовідведення надають вплив на якісні показники води; матеріал та діаметр труб; товщина стінок, герметичність, порушення стикових з'єднань, деформація тіла труб, геологія, наявність та агресивність ґрунтових вод, термін експлуатації; глибина закладання; динамічні навантаження на трубопровід; гідравлічний режим та інші. Усі пошкодження, викликають відмови на трубопроводах, обумовлені дією зовнішніх та внутрішніх факторів, що поділяються на три групи.

Перша група факторів пов'язана з технологією виготовлення труб. Для сталевих труб заводськими дефектами є металургійні. У чавунних трубах основним металургійним дефектом слід вважати раковини в тілі труб, що знижують їх несучу здатність.

Друга група факторів пов'язана з будівництвом трубопроводів. У чавунних трубопроводах це виражається в частих порушеннях стикових з'єднань, просіданнях ґрунту під трубами. У сталевих трубах це проявляється в їх переломі, порушенні зварних поперечних з'єднань. Слід

зазначити, що за індустріальних методів монтажу трубопроводів, особливо внутрішньоквартальних, повністю уникнути впливу другої групи факторів на надійність трубопроводів практично неможливо.

Третя група факторів визначається умовами функціонування та експлуатації трубопроводів. До цієї групи факторів відносяться матеріал та діаметр труб, ступінь зношеності трубопроводу та режим його роботи, тиск у мережі, агресивність транспортованого середовища, удари та ін. Наприклад, від такого фактора, як матеріал трубопроводу, на 18% залежить довговічність труб. У табл. 1.1 представлені дані про співвідношення основних видів ушкоджень на трубопроводах[9].

Таблиця 1.1 Структура основних видів ушкоджень на трубопроводах

Види повреждений	Материал труб				
	сталь	чугун	асбесто-цемент	пласт-масса	ЖБ
Нарушение стыковых соединений, %	15	70	80	90	60
Образование свищей, %	60	1	-	5	1
Разрыв труб и фасонных частей, %	2	2	2	5	-
Электрокоррозия, %	23	3	-	-	1
Перелом и разрушение труб, %	-	24	18	-	39

Система відведення стічних вод є найважливішим елементом благоустрою будь-якого населеного пункту. Вона складається з самопливних труб, колекторів, колодязів різного призначення, насосних станцій, напірних трубопроводів, очисних споруд. Цей комплекс інженерних споруд призначений для прийому, організованого відведення, очищення, знезараження та випуску стічних вод

у водойму. Одна з головних вимог до системи водовідведення – герметичність трубопроводів та надійність усіх її елементів.

Стійка робота мережі водовідведення забезпечується дотриманням правил технічної експлуатації та виконання заходів щодо її обслуговування. Здійсненням комплексного обслуговування водовідвідних мереж займаються спеціалізовані підприємства «Водоканали» із цілодобовою диспетчерською службою та аварійно-відновлювальними підрозділами. Ці підприємства оснащені спеціальним обладнанням для діагностики, ремонту та проведення профілактичних заходів щодо запобігання аварійним ситуаціям на мережах. Основними проблемами на мережах каналізації є збереження здатності гідравлічного транспорту стічних вод спрямована основна увага обслуговуючих організацій[5].

Методи обстеження каналізаційних систем розвиваються завдяки візуальним та інфрачервоним камерам, (сонарні / ультразвукові методи), каналізаційного сканування та технології оцінки (SSET), лазерного сканування локалізованого місця протікання (FELL) та радіолокатори, що проникають крізь землю (GPR). Інспекційні технології розвиваються паралельно досягненням у галузі сенсорних та комп'ютерних технологій, що полегшують та прискорюють збір та аналіз високоякісних даних[14].

Крім вищеперерахованих існують такі методи:

Лазерне профілювання (Laser Profiling) може надати дуже точні дані про форму та стан для внутрішньої стінки труби, що обстежується. Ці системи зазвичай використовуються для реєстрації прогину, деформації, овальності та зміни площі поперечного перерізу трубопроводу. Лазерне профілювання можна використовувати для оцінки руйнування стін (втрата товщини стінка труби). Апаратне забезпечення, що входить до комплекту або приєднане до камери, і кільцем лазерний промінь проходить на внутрішній поверхні труби, поки робот з камерою рухається трубопроводом. Після того, як всі лазерні зображення будуть захоплені та записані, супутнє програмне забезпечення використовує лазерні зображення, отримані при скануванні, та будує цифровий профіль труби стіна, яка була обстежена. Дані цифрового профілю можуть використовуватися для повідомлення даних, зібраних в декількох корисних форматах, таких як: графічні дані, тривимірне зображення та плоска складка зображення)[26].

Інструменти для вимірювання лазерних діодів (Laser Diode Measurement Tools) – використовується технологія вимірювання лазерних діодів при інспекції трубопроводів для вимірювання дефектів усередині труби під час звичайної відео інспекції (CCTV). Цей тип вимірювального інструменту використовується для вимірювання тріщин та розломів вимірювання, отвори в стиках та інші проблеми, що вимагають точного виміру, щоб надати інженеру правильну перспективу та точні дані для встановлення контрольних показників стану існуючої труби та прийнятності нової труби. Зазвичай головка камери оснащена двома лазерними

діодами. Лазерні діоди точно вирівняні на заводі і мають відому відстань (інтервал зазвичай десь від 2 до 3 дюймів або від 51 до 76 мм). Коли спостереження знаходиться в полі зору камери, яке необхідно виміряти, лазерна діодна система може бути використана для цього. Лазерний промінь випускається кожним діодом (який відображається у вигляді маленької червоної точки) на стінці труби, що обстежується. Ці лазери встановлюють еталонний калібрований точки, у яких можна точно виміряти характеристики та дефекти труби)[26].

Сонар (Sonar) - профілювальник сонара призначений для надання даних про розміри рівня засміченості, скупчення жиру, деформація труб та інші аномалії нижче рівня води, де візуальний огляд використовувати не можна. У додаткових лініях чи сифонах гідролокатор може забезпечити профільні та розмірні дані значних перешкод чи дефектів. Використовуючи спеціальне програмне забезпечення для сонара, кругове накладання, що становить вихідні розміри труби, проектується та переміщується для візуального та цифрового порівняння пошкоджень поверхні, відкладень або у деяких випадках товщина стінки. Програмне забезпечення сонара може використовуватися для вимірювання величини зміщення стику, рівнів засміченості, розміру засмічення, рівні скупчення мастила та інші дефекти. Сонар не надає інформацію вище за рівень води, але може використовуватися на частково затоплених лініях у поєднанні з системою відеоспостереження (CCTV)[26].

З досвіду експлуатації водовідвідних мереж одним з діючих і продуктивних методів спостереження є метод теледіагностування підземних комунікацій(ССТV) (див. рис. 1.2), який дозволяє досить точно встановити рівень технічного стану і характер пошкоджень, якщо вони є. З отриманих даних виробляються заходи, дозволяють максимально ефективно вирішити експлуатаційні завдання[5].



Рисунок 1.2 Метод теледіагностування підземних комунікацій(ССТV),

Пристрої для телеінспекції трубопровідних мереж можна класифікувати так[12]:

1) Проштовхувані системи телеінспекції : призначені для обстеження трубопроводів діаметром від 15 до 500 мм. Довжина кабелю таких пристроїв зазвичай не перевищує 100 м (див. рис. 1.3);



Рисунок 1.3 Проштовхувані системи телеінспекції

2) Системи телеінспекції з дистанційно керованими роботами: призначені для обстеження трубопроводів діаметром від 100 до 3000 мм (див. рис.к 1.4); довжина кабелю може досягати 600 м. окремих моделей роботів, здійснює зйомку, може бути представлений у форматі транспортного або плаваючого модуля. Перший застосовується для обстеження спорожнених трубопроводів (див. рис. 1.5), другий – для інспекції трубопроводів, частково заповнені водою (див. рис. 1.6);



Рисунок 1.4 Системи телеінспекції з дистанційно керованими роботами



Рисунок 1.5 Модульна телеінспекційна система
ROVION®
для труб діаметром 100 – 2000 мм



Рисунок 1.6 Плаваючий модуль Rausch W-350

3) Системи телеінспекції для бестраншейного ремонту: призначені для трубопроводів діаметром від 70 до 1600 мм, поділяються на прошовхувані (див. рис. 1.7).та роботизовані (див. рис. 1.8).



Рисунок 1.7 Роботизована ремонтна фрезерна система
IMS Robotics
DRIVE (Німеччина)



Рисунок 1.8 Автолабораторія з роботизованими
системами ремонту

та відновлення трубопроводів IMS Robotics (Німеччина)

Інспекція CCTV зазвичай включає в себе два основних процеси: збір відео на місці і оцінка відео за межами об'єкта. Процес збору даних зазвичай виконується технологами інспекції з дотриманням певного стандарту інспекційної операції, наприклад, RASR, який поширений на Північно Американському континенті, який увібрав в себе також британські стандарти [24]. За американським стандартом RASR дефекти каналізаційних труб підрозділяються на (див. рис 1.9):

1. Конструкційні;
2. Структурні;
3. Дефекти обслуговування;
4. Дефекти експлуатації;

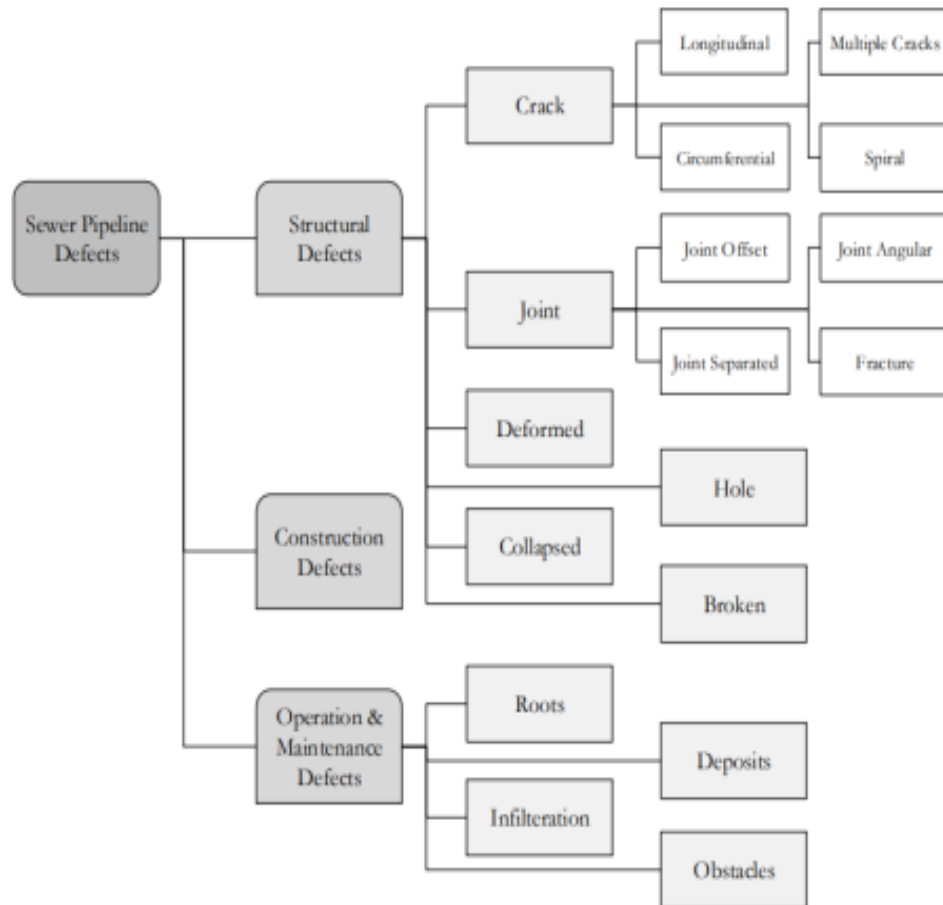


Рисунок 1.9 – Стандарт PACP: класи дефектів стічних труб [23, 25].

Стічна труба може втратити гідравлічну пропускну здатність через просіданням труби (Див. рисунок 1.10).

Спершу утворюється розрив, далі проникнення в трубу ґрунтових вод, так звана інфільтрація зі зникненням підтримки ґрунту навколо труби до утворення просадки та зміщення центру мас труб, яке нерівномірно навантажує стики труб без належної підтримки ґрунту, призводить до перенавантаження і руйнування труб.

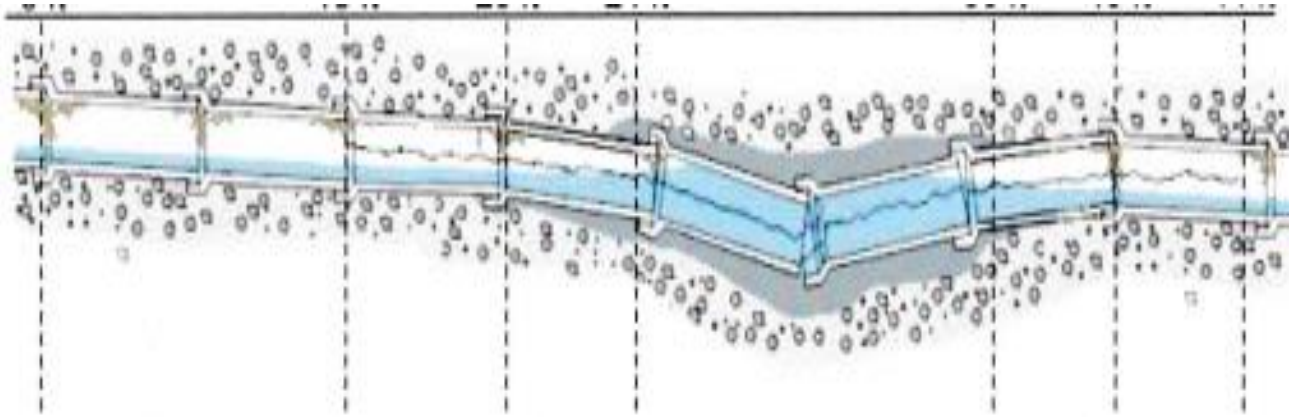


Рисунок 1.10 – Просідання труби після трьох етапів [22]

По мірі протікання процесу, тріщини чи розломи можуть ламатися та деформуватися. Камера труби затопляється по мірі підвищення рівня води. Також виникають проблеми, що унеможливають можливість повноцінного завершення інспекції через зруйнований сегмент труби. На рис. 1.11 показано де зміщення стиків труб ускладнює обстеження, а іноді і припиняє його через великі тріщини, уламки та розломи[26].

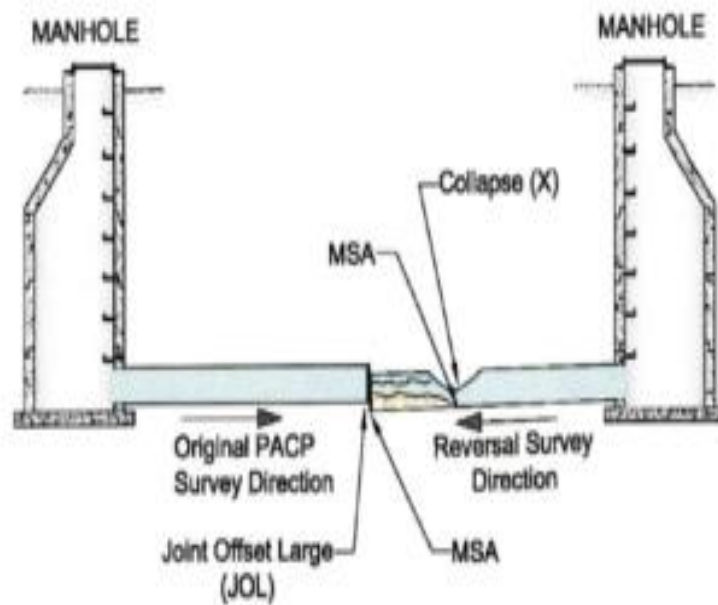


Рисунок 1.11 – Приклад зруйнованого стику труби, що припиняє інспекцію [26].

Не дивлячись на певні недоліки інспекцій CCTV такі, як професіоналізм оператора, рівня засміченості та якості зображення з ефектом освітлення, технологія CCTV поширені при намаганні експлуатаційних організацій отримати інформацію про функціональний стан каналізаційних трубопроводів. Як правило замовники інспекцій хочуть отримати звіти, що включають описи дефектів з візуальним підтвердженням для оцінки функціонального стану системи перед ухваленням різних рішень. Останні розробки в сфері технологій обробки візуальних даних призвели до різкого зниження собівартості інспектування. Цей факт разом з підвищенням розрахункових можливостей комп'ютерного обладнання робить результат роботи алгоритмів методів комп'ютерного зору і машинного навчання більш ефективним.

На рисунку 1.12 видно розповсюдження загальних методологій, що використовуються для аналізу результатів відео інспекцій по всьому світу за роками. Останні рки дослідники перевагу віддають методам Deep Learning та Machine Learning[22].

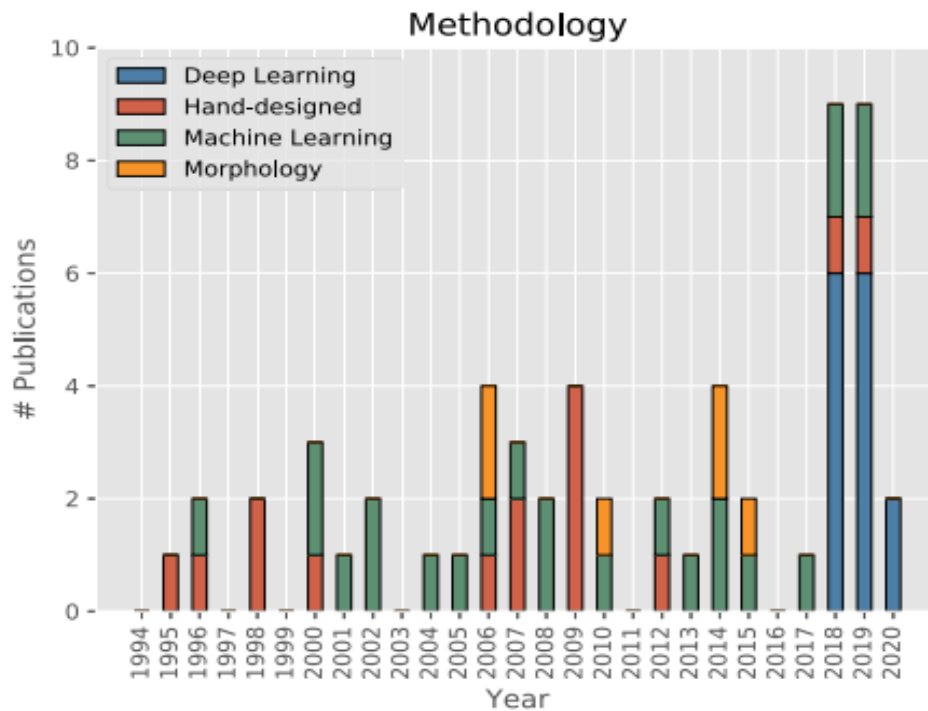


Рисунок 1.12 Розповсюдження загальних методологій, що використовуються для аналізу результатів CCTV/SSET

Розглянемо такі визначення для розуміння глибокого навчання порівняно з машинним навчанням та штучним інтелектом.

Глибоке навчання (Deep Learning) – це різновид машинного навчання на основі штучних нейронних мереж. Процес навчання називається глибоким, так як структура штучних нейронних мереж складається з кількох вхідних, вихідних та прихованих шарів. Кожен шар містить одиниці, що перетворюють вхідні дані у відомості, які наступний шар може використовувати для певної задачі прогнозування.

Завдяки цій структурі комп'ютер може навчатися за допомогою власної обробки даних.

Машинне навчання (Machine Learning) — це підмножина штучного інтелекту, у якому використовуються методи (наприклад, глибоке навчання), дозволяють комп'ютерам використовувати досвід удосконалення у вирішенні завдань. Процес навчання ґрунтується на наступних діях:

1. Передає дані в алгоритм.
2. Ці дані використовуються для навчання моделі.
3. Тестування та розгортання моделі.
4. Використання розгорнутої моделі для автоматизованого вирішення задачі на основі прогнозування.

За допомогою прийомів машинного навчання та глибокого навчання можна створювати комп'ютерні системи та програми, які виконують завдання, які зазвичай доручаються людям. До цих завдань належать розпізнавання зображень, розпізнавання мови та мовний переклад.

Подивимось на рисунок 1.13, де видно, які підходи використовує Deep Learning і Conventional Machine Learning [30].

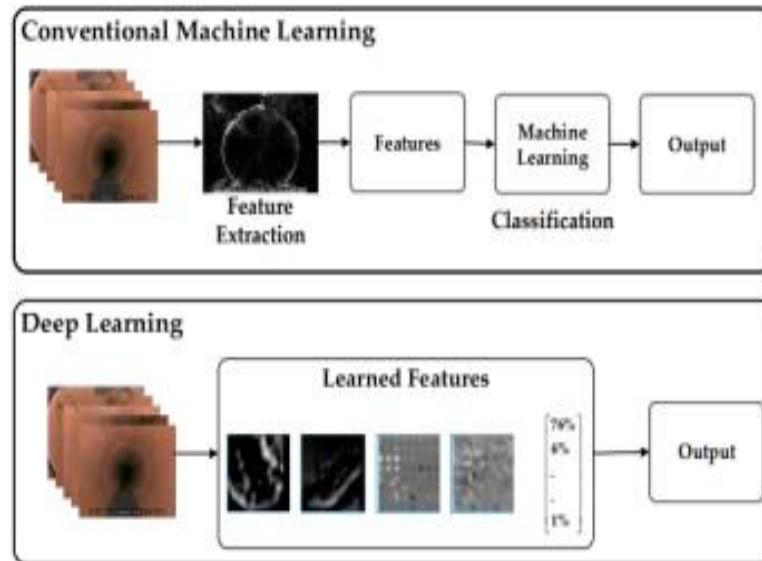


Рисунок 1.13 Ілюстрація різниці підходів в Machine Learning і Deep Learning[26].

Давайте порівняємо ці два методи. При машинному навчанні алгоритму необхідно повідомити, як виконувати точний прогноз, використовуючи додаткову інформацію (наприклад, шляхом отримання даних). У разі глибокого навчання алгоритм зможе навчитися, як створювати точний прогноз шляхом самостійної обробки даних за допомогою структури штучних нейронних мереж[13].

Табл. 1.2 Детальне порівняння Machine Learning і Deep Learning [46]

	Все машинне навчання	Тільки глибоке навчання
Кількість точок даних	Для створення прогнозів можна використовувати невеликі обсяги даних.	Необхідно використовувати великі обсяги навчальних даних для створення прогнозів.
Залежність від	Може працювати на	Залежить від високо-

обладнання	малопотужних комп'ютерах. Не потрібні великі обчислювальні потужності.	продуктивних комп'ютерів. При цьому комп'ютер виконує велику кількість операцій перемноження матриці. Графічний процесор може ефективно оптимізувати ці операції.
Процес конструювання ознак	Вимагає точного визначення ознак та його створення користувачами.	Розпізнає ознаки високого рівня на основі даних та самостійно створює нові ознаки.
Підхід до навчання	Процес навчання розбивається на дрібні кроки. Потім результати виконання кожного кроку поєднуються в єдиний блок вихідних даних.	Завдання вирішується шляхом наскрізного аналізу.
Час виконання	Навчання займає порівняно мало часу – від кількох секунд до кількох годин.	Як правило, процес навчання займає багато часу, оскільки алгоритм глибокого навчання включає багато рівнів.
Вихідні дані	Вихідними даними зазвичай є числове значення, наприклад, оцінка або класифікація.	Вихідні дані можуть мати кілька форматів, наприклад, текст, оцінка або звук.

1.2 Аналіз методів та моделей машинного навчання і комп'ютерного зору

Стрімкий розвиток технологій машинного навчання пов'язаний із зростанням обсягу цифрової інформації та потужності комп'ютерного обладнання. Глибоке навчання використовується під час вирішення комплексу складних

завдань: класифікації, сегментації та ідентифікації дефектів. Моделі із застосуванням методів глибокого навчання, таких як згорткові нейронні мережі (CNN), в основному використовуються в задачах семантичної сегментації та класифікації. Вони також знаходять своє застосування за необхідності збільшення продуктивності прогнозування на основі великого обсягу даних та наявності обчислювальних ресурсів. Проблеми, які раніше вважалися нерозв'язними, тепер вирішуються з більшою точністю[13].

Принцип роботи традиційних методів комп'ютерного зору полягає у вилученні векторів об'єктів із зображень для їх подальшої класифікації. Нейронні мережі, що використовуються при глибокому навчанні, навчаються, а не програмуються. Отже, за порівняно з традиційними методами комп'ютерного зору, вони вимагають меншого експертного аналізу. Глибоке навчання забезпечує гнучкість, за рахунок якої модель і структура нейронних мереж може бути перенавчена для будь-якого можливого набору даних, що критично важливо у завданнях ідентифікації дефектів. Дана можливість є перевагою, в порівнянні з алгоритмами комп'ютерного зору, що адаптуються під конкретні завдання[6].

У роботі [19] під керівництвом Andy Lee проведено порівняльний аналіз глибокого навчання та традиційних методів комп'ютерного зору на прикладі розпізнавання об'єктів. Автори спорудили робота, оснащеного технологіями OpenCV і Caffe. В результаті було виявлено, що згорткові нейронні мережі показали найкращу продуктивність, проти традиційними методами комп'ютерного зору. Результати цієї роботи можуть бути використані при проектуванні механічного робота, який буде використовуватися для ідентифікації дефектів на різних поверхнях.

Зі зростанням обчислювальних потужностей графічних процесорів стало можливим застосовувати глибші архітектури моделей машинного навчання [41]. З розвитком згорткових нейронних мереж стала доступна можливість більш ефективно вивчати та узагальнювати особливості зображень. З їхньою допомогою вирішуються завдання класифікації [42], ідентифікації образів [43], зокрема, транспортні завдання.

Традиційний підхід комп'ютерного зору виявлення дефектів полягає у використанні таких методів як SIFT (Scale-Invariant Feature Transform), SURF (Speeded-Up Robust Features) та BRIEF (Binary Robust Independent Elementary Features). На етапі видалення ознак у задачах класифікації зображення можуть бути задіяні такі алгоритми:

- ❖ детектор меж Канні – використовує поетапний алгоритм виявлення широкого діапазону країв у зображеннях;
- ❖ детектор кутів Харріса – застосовується для вилучення кутів та отримання властивостей зображення;
- ❖ порогова сегментація – цей алгоритм може бути використаний для отримання бінарного зображення[19].

Складність цього підходу у тому, що кожного зображення треба вибрати, які ознаки слід враховувати. У міру збільшення кількості класів, вилучення ознак стає все більш громіздким, від чого конфігурація класів об'єктів стає тривалою дією, що призводить до підвищеного ризику виникнення помилок.

Завдяки розвитку апаратного забезпечення та збільшення граничної обчислювальної потужності розробка згорткових нейронних мереж справила величезний вплив на вирішення завдань розпізнавання дефектів.

Назву архітектура мережі отримала через наявність операції згортки, суть якої полягає в тому, що кожен фрагмент зображення множиться на матрицю (ядро) згортки поелементно, а результат підсумовується та записується в аналогічну позицію вихідного зображення. Поряд з цим, глибоке навчання представляє концепцію наскрізного навчання, що перед-

бачає передачу машині набору анотованих зображень з інформацією про те, які класи об'єктів присутні на вхідному наборі даних. Таким чином, модель глибокого навчання навчається. Глибокі нейронні мережі виявляють базові шаблони класах зображень та автоматично виробляють найбільш наочні та характерні особливості щодо кожного конкретного класу кожного об'єкта. Звідси виходить більш висока продуктивність і точність порівняно з традиційними алгоритмами машинного зору[19].

Штучні нейронні мережі формуються за допомогою шарів зв'язаних вузлів. У моделях глибокого навчання використовуються нейронні мережі з великою кількістю рівнів. Розглянемо найпопулярніші типи штучних нейронних мереж[47].

Нейронна мережа з передачею по черзі – це найпростіший тип штучної нейронної мережі. У мережі з передачею по черзі інформація переміщається лише одному напрямку від вхідного рівня до вихідного. Нейронні мережі з передачею по черзі перетворюють вхідні дані, пропускаючи через кілька прихованих шарів. Кожен шар складається з набору нейронів та повністю з'єднаний з усіма нейронами у попередньому шарі. Останній повністю з'єднаний шар (вихідний шар) є висновком створених прогнозів[47].

Рекурентні нейронні мережі (RNN) - це штучні нейронні мережі, що широко використовуються. Ці мережі зберігають вихідні дані шару і передають його на вхідний шар, щоб поліпшити прогнозування на виході конкретного шару. У рекурентних нейронних мережах відмінні можливості для навчання. Вони широко використовуються для виконання складних завдань, таких як прогнозування тимчасових рядів, навчання розпізнавання рукописного введення та розпізнавання природного мовлення[47].

Перетворювачі - це архітектура моделі, яка підходить для вирішення проблем, що містять такі послідовності, як текст або дані часових рядів. Вони складаються з шарів кодувальника та декодера. Кодувальник приймає вхідні дані та зіставляє їх з числовим поданням, що містить певні відомості, наприклад контекст. Декодер використовує інформацію з кодувальника для отримання вихідних даних, наприклад, перекладеного тексту. Перетворювачі відрізняються від інших архітектур, що містять кодувальники та декодери, своїми вкладеними шарами уваги. Увага: метод концентрації на конкретних частинах вхідних даних на основі важливості їхнього контексту щодо інших вхідних даних у послідовності. Наприклад, при підсумовуванні статей новин не всі пропозиції важливі для опису основної ідеї. Якщо зосередитись на ключових словах у статті, формування зведених даних може бути зроблено в одному реченні – у заголовку.

Перетворювачі використовуються для вирішення проблем обробки природної мови, таких як переклад, створення тексту, відповіді на питання та формування зведених даних тексту[47].

Генеративно-змагальні мережі (GAN) – це регенеративні моделі, навчені для створення реалістичного вмісту, наприклад, зображень. Кожна така мережа складається з двох мереж, відомих як генератор та дискримінатор. Обидві мережі навчаються одночасно. Під час навчання генератор використовує випадкові перешкоди створення нових штучних даних, які схожі на реальні дані. Дискримінатор приймає вихідні дані генератора як вхідні дані та використовує реальні дані, щоб визначити, чи є створений вміст реальним або штучним.

Кожна з мереж конкурує одна з одною. Генератор намагається створити штучне вміст, який відрізняється від реального вмісту, тоді як дискримінатор намагається правильно класифікувати вхідні дані або як реальні, або як штучні. Потім вихідні дані використовуються для оновлення ваги обох мереж, щоб допомогти їм краще досягти відповідних цілей[47].

Генеративно-змагальні мережі використовуються для вирішення таких проблем, як перетворення зображень на зображення та прогресу віку.

Згорткова нейронна мережа (CNN) - це особливо ефективна штучна нейронна мережа, що має унікальну архітектуру. Шари в ній організовані у трьох вимірах: ширина, висота та глибина. Нейрони в одному шарі поєднуються не з усіма нейронами в наступному шарі, а лише з невеликою областю нейронів цього шару. Остаточний результат скорочується до одного вектора оцінки ймовірності, впорядкованого по глибині одного з вимірювань.

Згорткові нейронні мережі використовуються в таких областях, як розпізнавання відео, розпізнавання зображень та системах вироблення рекомендацій[47].

На рис. 1.14 зображена загальна архітектура CNN для класифікації зображень.

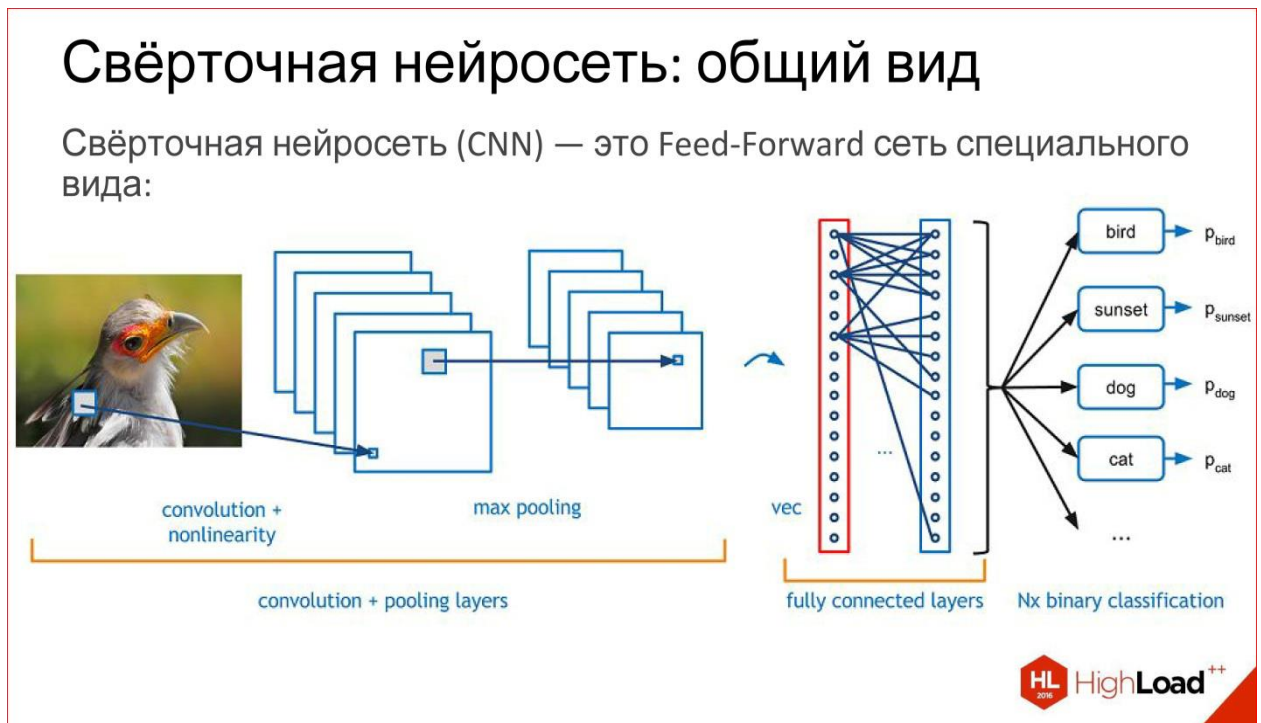


Рисунок 1.14 – Згорткова нейронна мережа: загальний вигляд, класифікація зобрження [3].

Згорткові нейромережі вирішують 3 основні завдання:

1. Класифікація. Ви подаєте картинку, і нейромережа просто говорить - у вас картинка про собаку, про коня, ще про щось, і видає клас.
2. Детекція - це найбільш просунуте завдання, коли нейромережа не просто говорить, що на малюнку є собака або кінь, але знаходить ще Bounding box - де це знаходиться на малюнку.
3. Сегментація. Тут ми говоримо про кожен піксель зображення: цей піксель відноситься до собаки, цей - до коня, а цей ще чогось. Насправді, якщо ви вмієте вирішувати задачу сегментації, то решта 2 завдань вже автоматично дано[3].

Що таке згорткова нейромережа? Насправді згорткова нейромережа - це звичайна Feed-Forward мережа, просто вона трохи спеціального виду. Що є в згортковій мережі? У неї є:

- 1) Згорткові шари - зазвичай малюються як набору площин чи обсягів. Кожна площина на такому малюнку або кожен зріз у цьому обсязі - це, по суті, один нейрон, який реалізує операцію згортки. По суті, це матричний фільтр, який трансформує вихідне зображення на якесь інше, і це можна робити багато разів.

- 2) Subsampling, або Pooling-шари, які зменшують розмір зображення. Шари субдискретизації просто зменшують розмір зображення: було 200×200 px, після Subsampling стало 100×100 px. По суті, усереднення трохи хитріше;
- 3) Звичайні повнозв'язні шари, той самий багат шаровий перцептрон, який просто зверху навішується на ці перші 2 хитрих шари[3].

Насправді згорткові шари працюють на обсягах. Тобто навіть якщо взяти звичайне зображення RGB, там вже 3 канали – це, по суті, не площина, а об'єм з трьох, умовно, кубиків. Згортка в цьому випадку вже буде не матрицею, а тензором - кубиком насправді.

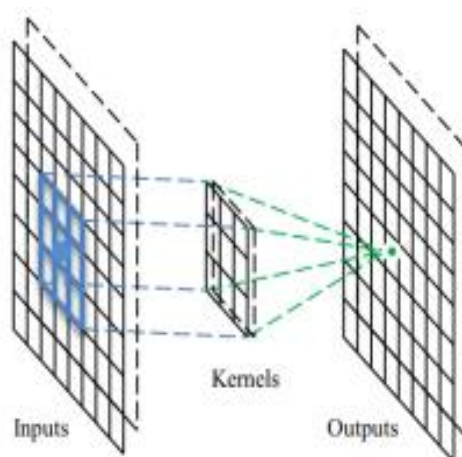


Рисунок 1.15 – Згортковий шар в згортковій нейронній мережі

У вас є фільтр, ви пробігаєте по всьому зображенню, він відразу дивиться на всі 3 кольорові шари і генерує одну нову точку для одного цього обсягу. Пробігаєте по всьому зображенню – побудували один канал, одну площину нового зображення. Якщо у вас 5 нейронів, ви побудували 5 площин. Так працює згортковий шар (див. рисунок 1.15). Завдання навчання згорткового шару - це завдання така ж, як у звичайних нейромереж - знайти ваги, тобто фактично знайти ту саму матрицю згортки, яка повністю еквівалентна вагам в нейронах. Що роблять такі нейрони? Вони фактично вчаться шукати якісь фічі, якісь локальні ознаки у тій невеликій частині, яку вони бачать – і все. Прогін одного такого фільтра - це побудова певної карти знаходження цих ознак у зображенні. Потім ви побудували багато таких площин, далі використовуєте їх як зображення, подаючи на наступні входи[3].

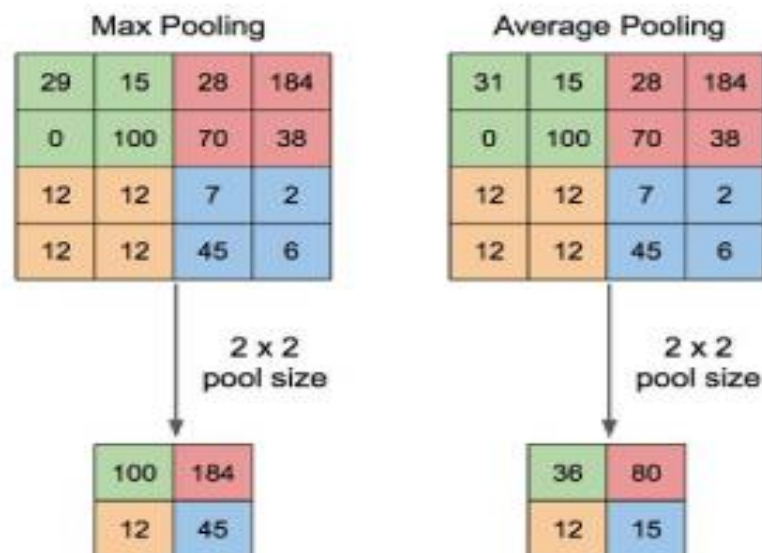


Рисунок 1.16 – Операція Pooling: усереднення чи взяття максимуму

Операція Pooling – ще більш проста операція. Це просто усереднення чи взяття максимуму (див. рисунок 1.16). Вона також працює на якихось невеликих квадратах, наприклад, 2×2 . Ви накладаєте на зображення i , наприклад, вибираєте максимальний елемент із цього квадрата 2×2 , відправляєте на вихід.

Таким чином ви зменшили зображення, але не хитрим Average, а трохи більш просунутою штукою - взяли максимум. Це дає невелику інваріантність до усунення. Тобто вам не важливо, яка ознака знайшлася в цій позиції або на 2 px вправо. Ця штука дозволяє неймережі бути трохи стійкішою до зсувів зображення[3].

Так працює Pooling-шар. Є кубик якогось обсягу - 3 канали, 10, або 100 каналів, які ви нарахували згортками. Він просто зменшує його по ширині та за висотою, решта розмірності не чіпає.

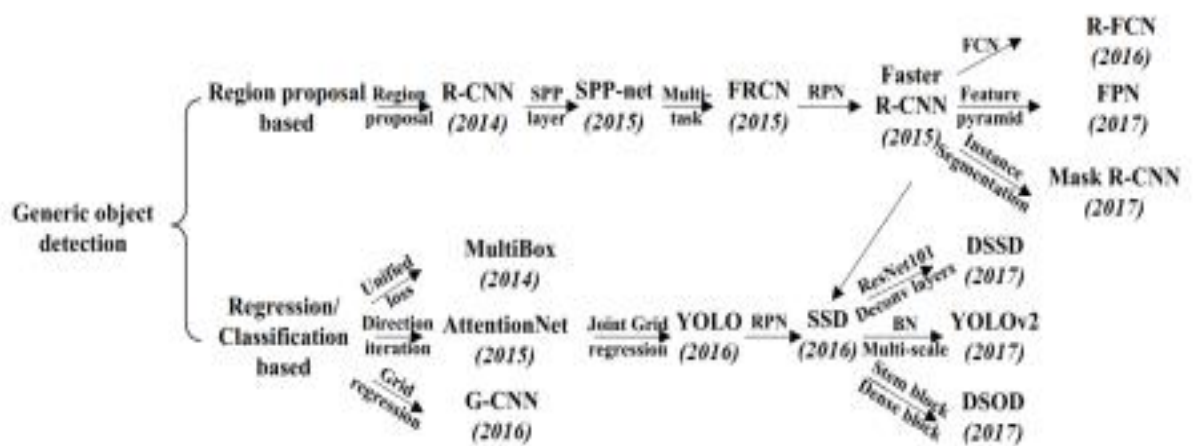


Рисунок 1.17 – Розвиток CNN, що розпізнають зображення [30].

Розвиток нейронних згорткових мереж, показана на рис. 1.17 призвів до появи трьох архітектур детектування об'єктів, що мають велику популярність: SSD, Faster R-CNN, YOLO.

Недоліками згорткових нейронних мереж є потреба, в деяких випадках, великому наборі даних та висока варіативність параметрів мережі, які суттєво впливають на результат роботи.

До таких параметрів відносяться: кількість шарів згортки, потреба в шари субдискретизації, кількість ядер і крок зсуву при обробці шару. Дані параметри вибираються емпірично. Глибоке навчання використовується для класифікації завдань, в яких широкий діапазон потенційних сигналів має бути відображено на обмежену кількість категорій, враховуючи, що є Достатній набір даних, внаслідок чого погано працює з відеопотоком та 3D-моделюванням[13].

1.3 Постановка задачі

В результаті аналізу актуальності проблеми був проведений короткий огляд інформації, який відкрив сучасні підходи в інспектуванні каналізаційних мереж. Також аналіз методів та моделей машинного навчання і комп'ютерного зору, показав можливе застосування методів глибоких нейронних мереж в процесі відео інспекцій стічних труб, а саме можливу автоматизацію процесу аналізу відео даних. Можна додати, що особливу увагу слід приділяти рівню води та місцям стику труб в трубах при діагностиці каналізаційних систем.

Дуже важливим для нашого міста і країни в цілому є підтримання на належному якісному рівні підземної інфраструктури з регулярним оцінюванням цих мереж.

Враховуючи вищесказане постановкою задачі можна вважати розробку інтелектуальної технології розпізнавання цілісності та просадки стічних труб на основі згорткової нейромережі.

Для виконання поставлених задач необхідно:

1. вибрати модель нейромережі для розпізнавання цілісності стічних труб;
2. вибрати модель нейромережі для розпізнавання рівня води в стічній трубі;
3. визначити методи навчання системи розпізнавання;
4. визначення та використання критеріїв оцінки якості роботи системи;
5. отримання результатів машинного навчання нейромережі;

2. МОДЕЛІ ТА МЕТОДИ АНАЛІЗУ ВІДЕОІНСПЕКЦІЇ СТІЧНИХ ТРУБ

2.1 Модель нейромережі для розпізнавання цілісності стічних труб

Здатність згорткових нейронних мереж (*convolutional neural network, CNN*) в різних напрямках розпізнавання, локалізації об'єктів і класифікації зображень, підштовхує розглядати їх, як основних кандидатів, в використанні аналізу результатів інспектувань каналізаційних систем. В наш час є ряд досліджень, спрямованих на використання досягнень в глибокому навчанні для автоматизації процесу винесення рішення щодо функціонального стану різних об'єктів міської інфраструктури. Але, застосування згорткових нейронних мереж (CNN) не таке розповсюджене в створенні інтелектуальних систем для інспекції каналізаційних мереж [29]. Також алгоритми глибоких нейронних мереж потребують великих обчислювальних ресурсів.

Модифікація MobileNet → MobileNetV2 має певні відмінності від традиційної структури згорткових нейронних мереж (*convolutional neural network, CNN*) (див. рис. 1.14), ця відмінність полягає в розділеному блоку згортки на різні шари (див рис. 2.1, рис. 2.2, рис. 2.3).

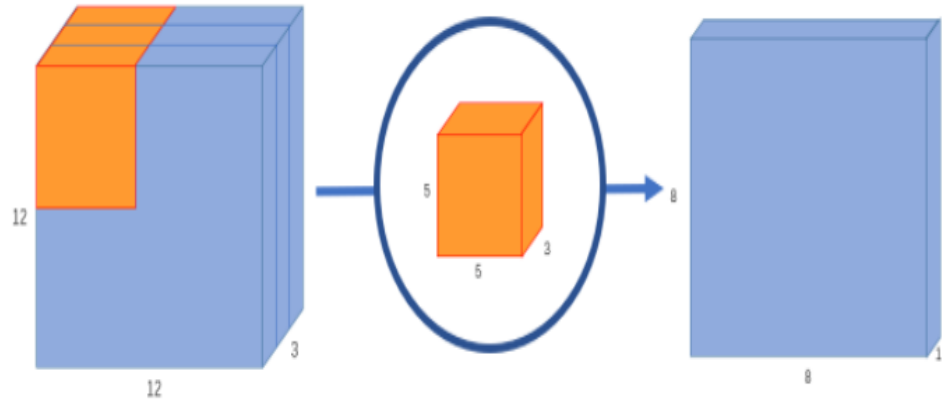


Рисунок 2.1 Розділення блоку згортки на різні шари

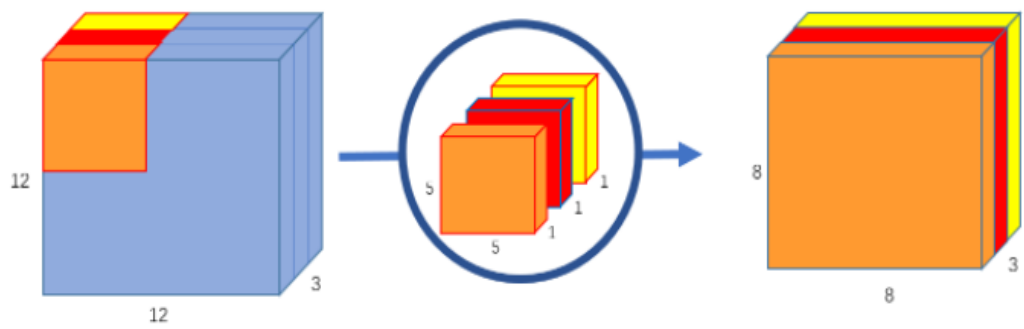


Рисунок 2.2 Перший шар згортка за глибиною

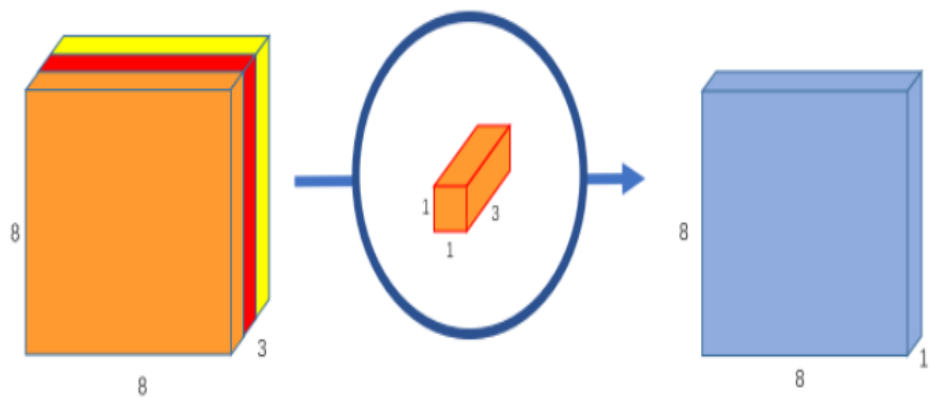


Рисунок 2.3 Другий шар точкової згортки

Архітектура MobileNetV2 при вирішенні практичних задач детектування об'єктів показала себе точною, надійною і легкою моделю згортової нейронної мережі. Тому використаємо переваги цієї архітектури для створення моделі нейромережі, яка буде детектувати функціональний стан стічної труби, який буде складатися з таких рівнів алфавіта класів класифікатора дефектів, як:

- 1) JointOffsets – Зміщення з'єднання;
- 2) Deposits – Жирові поклади;
- 3) Obstacles – Перепони;
- 4) Roots – Коріння.

Схематично ця нейромережа буде мати вигляд такий (див. рис. 2.4):

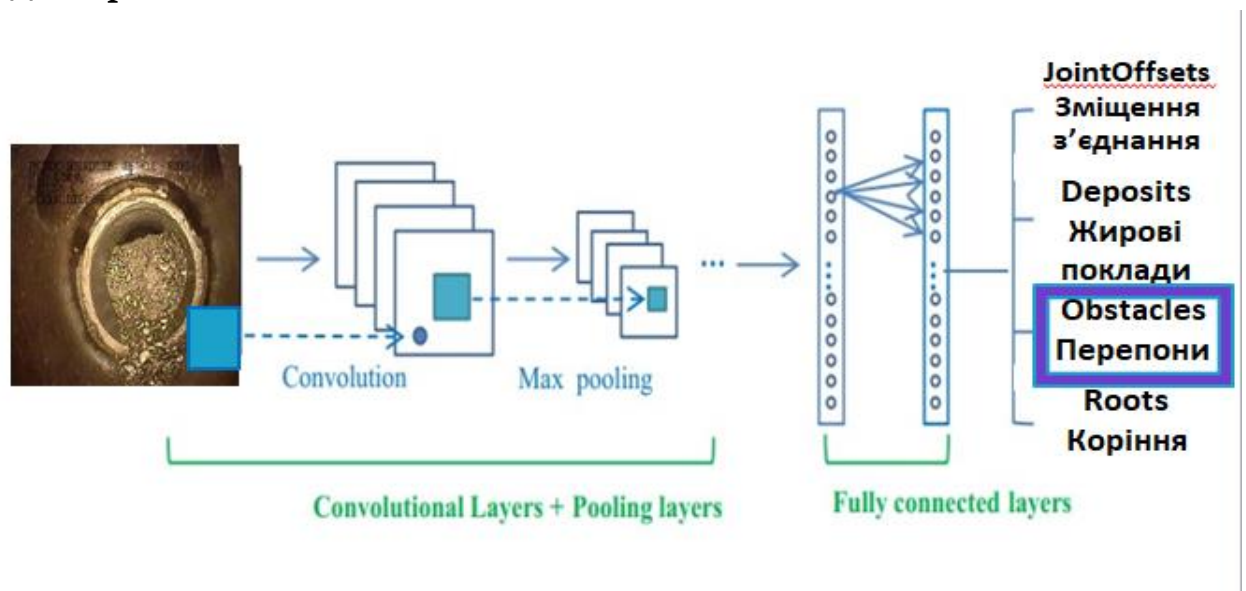


Рис. 2.4 Схематичне зображення моделі згортової нейронної мережі по розпізнаванню функціонального стану стічних труб

2.2 Модель нейромережі для розпізнавання рівня води в стічній трубі

Для створення даної моделі нейромережі також використаємо згортові нейронні мережі з використанням модифікації MobileNetV2.

Данна модель буде розпізнавати рівень води в трубі, а саме реагувати на підвищення чи зменшення цього рівня при відсутності дефектів, які розпізнає перша модель неймережі, що призначена слідкувати за цілісністю сканалізаційної труби.

Для цієї моделі виберемо алфавіт класів класифікатора дефектів, який буде складатися з таких рівнів (Water level): 0 - 5 - 10 - 15 - 20 - 25 - 30 - 35 - 40 - 45 - 50 - 55 - 60

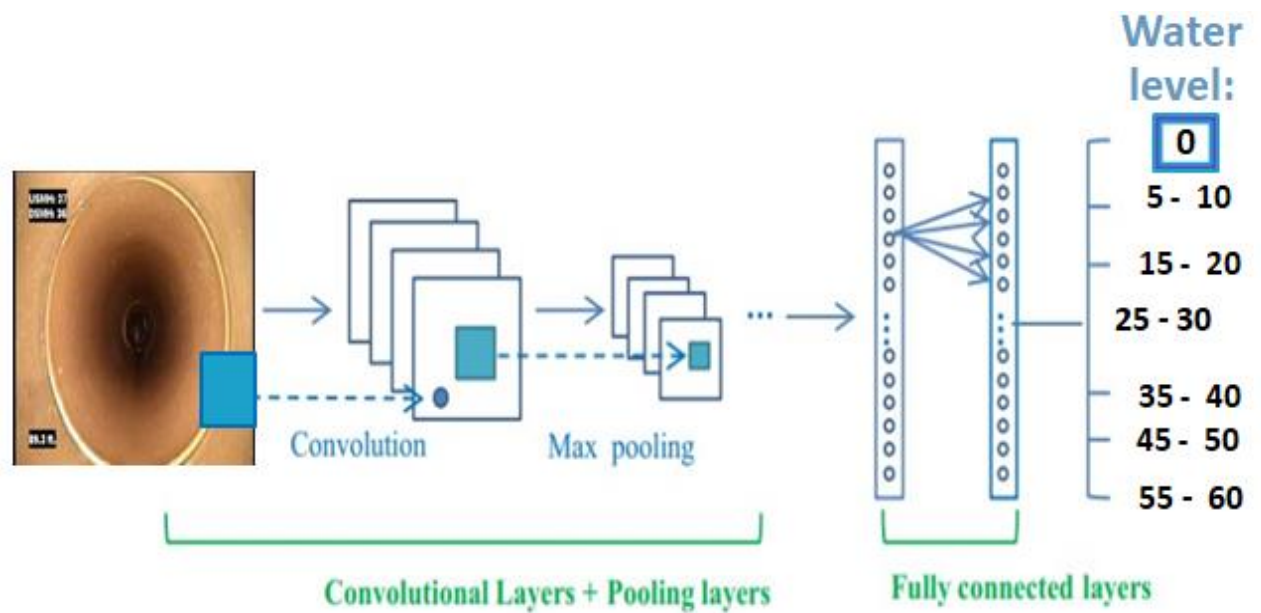


Рис. 2.5 Схематичне зображення моделі згорткової нейронної мережі по розпізнаванню рівня води стічної труби

2.3 Методи навчання системи розпізнавання

Перед навчанням моделей штучного інтелекту сперше необхідно якісно підготувати навчальні дані. Просто підготувати вже не достатньо треба ще ці данні модифікувати для розширення самого набору. Цей процес, що називається аугментація здійснюється через додавання модифікованих оригінальних зображень. Якщо виборка даних обмежена, то їх додатково ще модифікують Модель,

яка навчалася на точних даних, може не правильно розпізнавати кадри з різними не точностями.

При аугментації навчальну вибірку зображень змінюють по різному

- додавання шуму, розмиття;
- відрізання окремих частин самих зображень;
- розтягування та інше.

Проте щоб не створити зайвих проблем модифікацію зображень слід проводити після аналізу поставленої задачі.

Як правило наявні дані ділять на навчальні та тестові вибірки. Тестовий набір під час навчання не використовується. Навчальний і тестовий набори даних повинні розділятися, щоб класи мали однакову кількість даних. Для цього використовують стратифіковане розбиття.

Проте після такої оптимізації гіперпараметрів можна втратити узагальнюючі чинники. Тому на допомогу приходять нормалізація (normalization). Мета цього етапу приведення до єдиного виду різних даних з різними одиницями вимірів та діапазонів до певної схожості чи певної можливості поєднань.

Користуючись нормалізацією можна домогтися «спрощення ландшафту» функцій, що призведе до пришвидшення навчання, мало буде локальних мінімумів та швидше будуть знаходити мінімум[5]. Стандартизація (standardization) часто використовується як метод нормалізації, оскільки вона виявилася дуже ефективною.

Проте нормалізація передбачає зміну діапазону в даних без зміни форми розподілу, а стандартизація змінює форму розподілу даних (приводиться до нормального розподілу). Стандартизація у вигляді формули:

$$x' = \frac{x - \bar{x}}{\sigma} \quad (2.1)$$

де x – початковий вектор ознак, \bar{x} – середнє значення вектору ознак, а σ – стандартне відхилення.

2.4 Критерії оцінки якості роботи системи

Втрата середньої квадратичної помилки – це функція втрат регресії. MSE обчислюється як середнє квадратичне відхилення між передбаченими та фактичними значеннями. Незалежно від того, який знак мають прогнозовані та фактичні значення, ви завжди отримуватимете позитивний результат. Ідеальним значенням буде 0,0. Незважаючи на те, що ви можете зробити негативний результат та використати значення втрат для процесу оптимізації максимізації, результат буде мінімальним.

Перехресну втрату ентропії можна назвати логарифмічною втратою, перехресною ентропією, логарифмічною втратою або логістичною втратою. Вона показує, що кожна ймовірність, яку ви пророкуєте, порівнюється з фактичним значенням виходу класу як 0 або 1. Ця техніка обчислює бал. Ця оцінка штрафує ймовірність через відстань між вихідним значенням та очікуваним значенням. Характер штрафу буде логарифмічним. Велика різниця міститиме величезну оцінку, наприклад, 0,9 або 10. Однак, менші відмінності будуть містити невеликі бали, такі як 0,1 або 0,2.

Модель з точними ймовірностями міститиме логарифмічну втрату або крос-ентропію, що дорівнює 0,0. Це

вказує на те, що втрати крос-ентропії є мінімальними, і менші значення будуть представляти хорошу модель, а не великі. Серед усіх прикладів, крос-ентропія для двокласового передбачення чи бінарних завдань обчислюватиме середню крос-ентропію[23].

3. РЕАЛІЗАЦІЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ РОЗПІЗНАВАННЯ ЦІЛІСНОСТІ ТА ПРОСАДКИ СТІЧНИХ ТРУБ

3.1 Формування навчальних та тестових даних

Для побудови першої моделі нейромережі для розпізнавання цілісності стічних труб було сформовано відповідний набір навчальних та тестових даних зразки приведені(див. рис 3.1):



Рис. 3.1 Зразки тестових та навчальних даних для першої моделі

Для побудови другої моделі нейромережі для розпізнавання рівня води в стічній трубі також сформували відповідний набір навчальних та тестових даних зразки приведені(див. рис 3.2):



Рис. 3.3 Зразки тестових та навчальних даних для першої моделі

3.2 Результати машинного навчання нейромережі

Результати навчання моделі нейромережі для розпізнавання цілісності стічних труб приведені на Рис. 3.4:

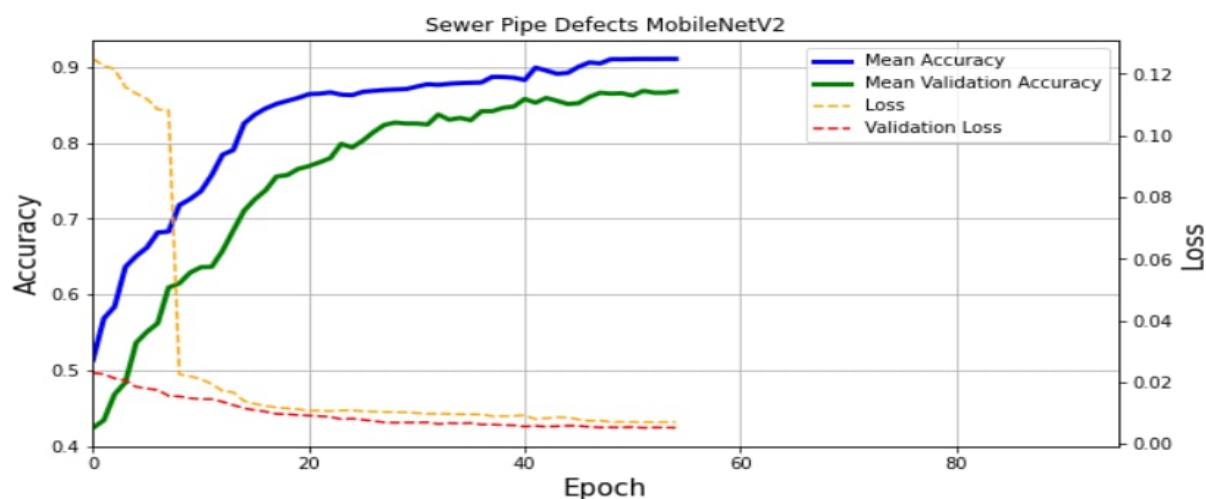


Рис. 3.4 Результати навчання першої моделі нейромережі

За результатами навчання видно, що модель має точність 80 % за тестовою навчальною матрицею.

Результати навчання моделі нейромережі для розпізнавання цілісності стічних труб приведені на Рис. 3.5:

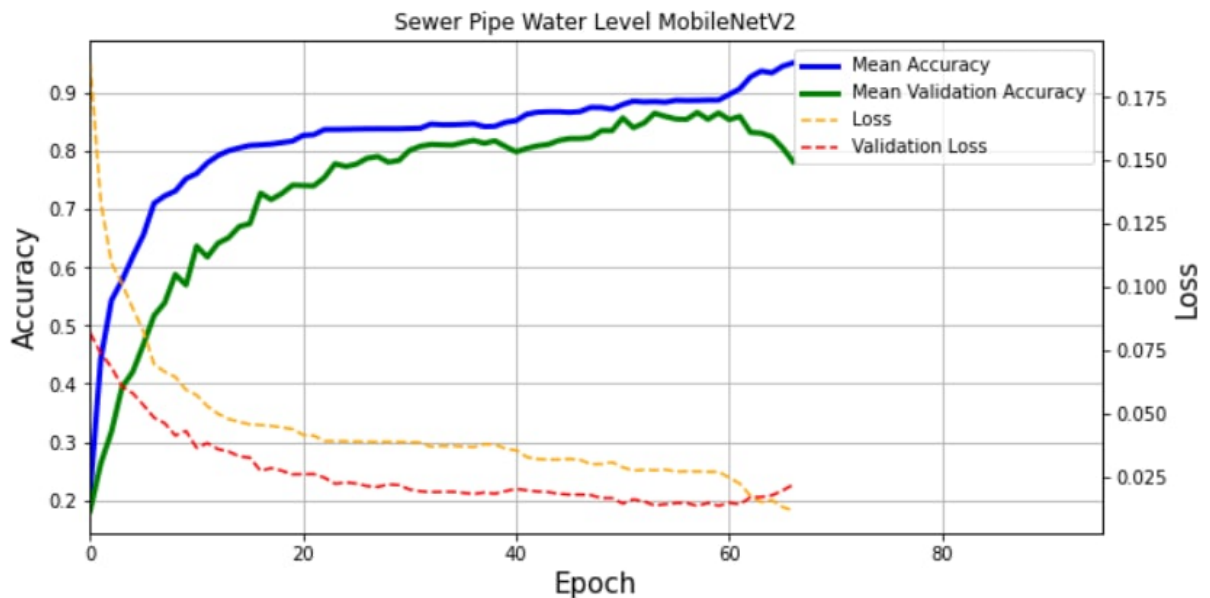


Рис. 3.5 Результати навчання другої модулі нейромережі

За результатами навчання видно, що модель має точність 80 % за тестовою навчальною матрицею.

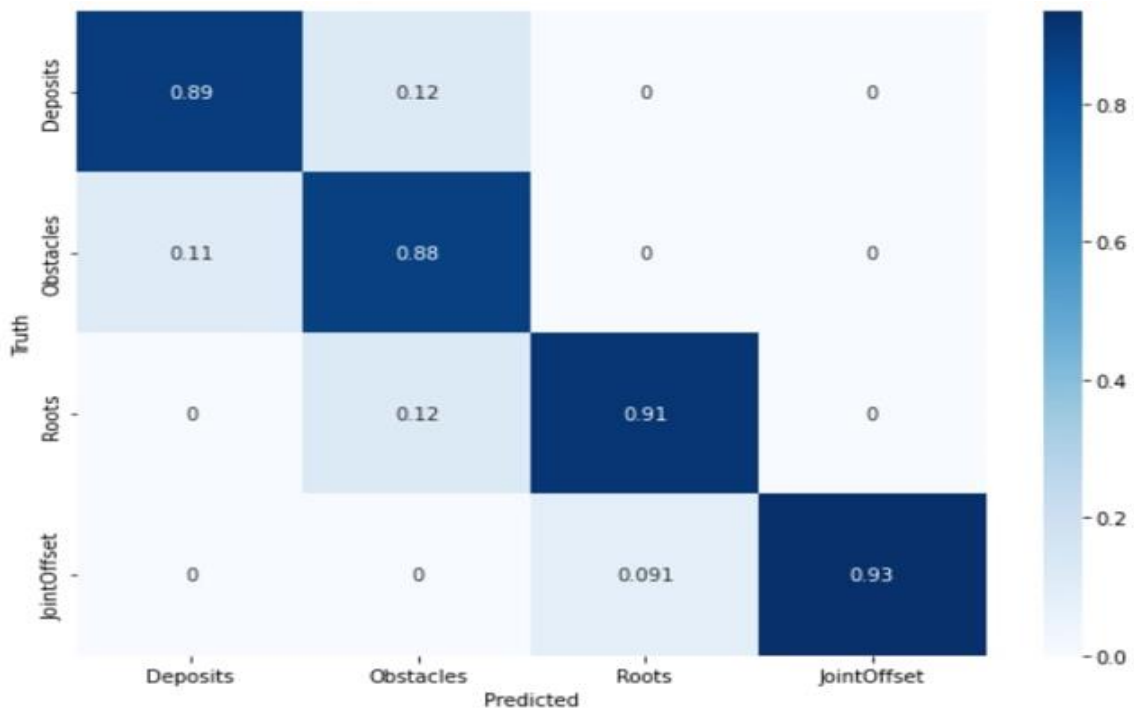


Рис. 3.6 Матриця помилок першої моделі неромережі

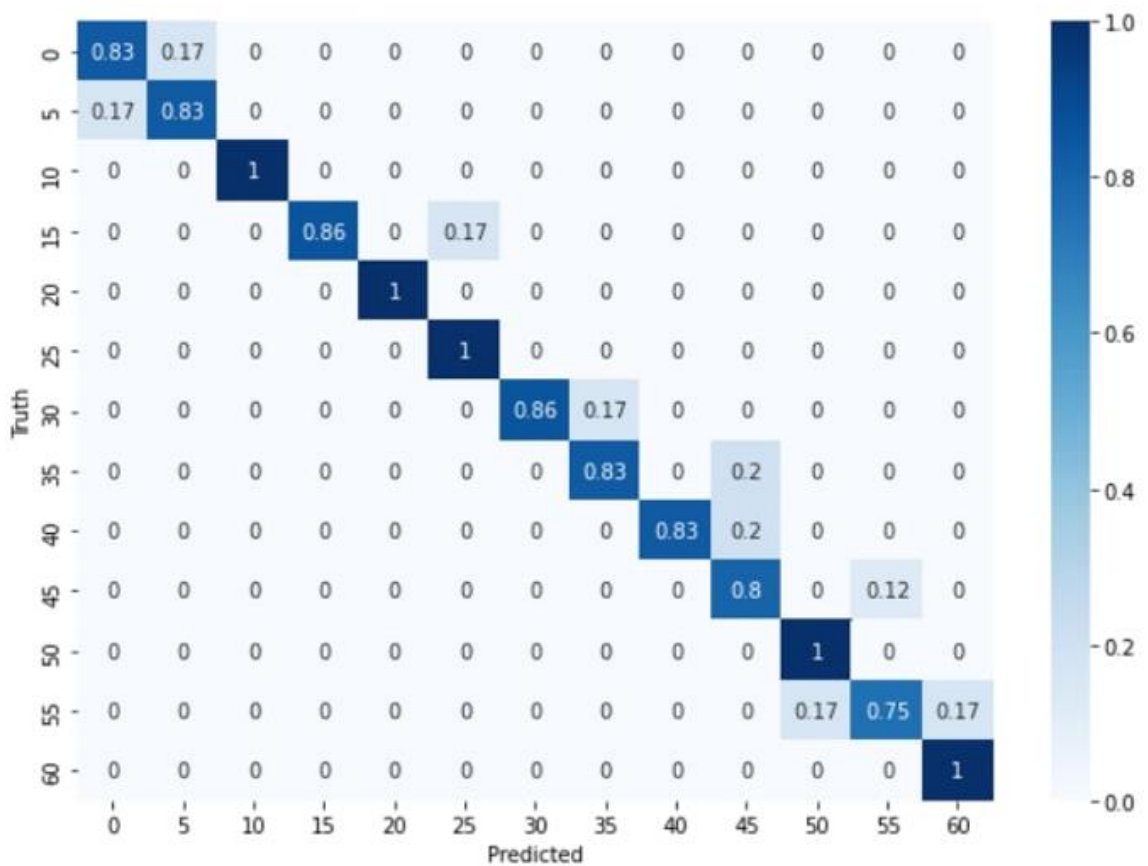


Рис. 3.6 Матриця помилок другої моделі неромережі

Також відбувалося тестування на реальних даних.

3.3 Опис алгоритму роботи інтелектуальної системи

Інтелектуальна система обробляє відео двома моделями. Кожна модель шукає свої класи від кадру к кадру, здійснює по кадрава обробка відео. Якщо спрацьовує перша модель згорткової нейронної мережі, друга – не включається. Коли змінюється рівень води без спрацювань першої моделі, то починає працювати алгоритм другої моделі загорткової неронної мерержі, який шукає місця початку і кінця просадки, використовуючи показання рівня води.

3.4 Програмна реалізація

Програма була реалізована на мові програмування Python з використанням відомих фреймворків Tenderflow і Keras

ВИСНОВКИ

Робота присвячена аналізу методів прогнозування показників функціонального стану труб водовідведення. Дана проблема підвищення точності і якості виявлення та прогнозування функціонального стану стічних труб є актуальною.

Метою роботи є аналіз та синтез інформаційних технологій для виявлення певного набору дефектів на відеозаписах зроблених інспекцією CCTV.

Методом оцінки стану каналізаційної труби є використання CCTV. Визначені предмет та об'єкт дослідження.

Проведений аналіз особливостей сучасних підходів в інспектуванні каналізаційної мережі, встановлені параметри, які впливають на якість та надійність такої діяльності. В роботі проведений огляд методів та моделей машинного навчання і комп'ютерного зору, які можуть бути застосовані для отримання актуальних даних. Визначено, що найбільш прийнятним є апарат штучних нейронних мереж.

В результаті були реалізовані дві моделі штучних згорткових нейронних мереж: перша - модель нейромережі для розпізнавання цілісності стічних труб та друга - модель нейромережі для розпізнавання рівня води в стічній трубі, які спрацьовують по черзі. Тестування моделей на навчальній та реальних вибірках показали задовільні результати.

СПИСОК ЛІТЕРАТУРИ

1. Абдурасулов И. Водообеспечение и очистка сточных вод Кыргызской Республики. Монография. Часть 1,2.– Бишкек: Илим,1993. – 449 с.
2. Алейнікова, А.І. Методологічні основи подовження експлуатаційного ресурсу підземних інженерних мереж [Текст] / А.І. Алейнікова, В.М.Волков, Д.Ф. Гончаренко, Г.Г.Зубко, О.В. Старкова: під заг.ред. Старкової О.В. – Х.: Раритеты Украины, 2017. – 320 с.
3. Бунин О. Введение в архитектуры неронных сетей. – 16.10.2017. <https://habr.com/ru/company/oleg-bunin/blog/340184/>
4. Крилова І. І. Аналіз сучасного стану сфери водопостачання та водовідведення в Україні / І. І. Крилова // Інвестиції: практика та досвід. – Київ, 2018. – С. 118 - 125
5. Коплик А.В. Магістерська робота на тему: «Інтелектуальна система оцінки функціонального стану стічних труб». <https://essuir.sumdu.edu.ua/bitstream-download/123456789/85534/1/>
6. Корпук_бас_роб.pdf;jsessionid=FF2F9BC62F34B37AF82504BADEB2C437
7. Косухин А.М. Водотводящие инженерные сети в прошлом, настоящем и будущем / Вестник БГТУ им. В.Г. Шухова. – 2017. - № 6 – С. 29 - 34
8. Кравченко О.В. Сучасний стан мереж водопостачання та водовідведення України: проблеми та перспективи розвитку: [текст] / О.В. Кравченко, О.Ю. Ямко // 3б. доп. між. Конгресу та технічної виставки «ЕТЕВК-2017». – Чорноморськ, 2017. – С. 86–89
9. Мызникова М. А. Современное состояние и тенденции развития жилищно-коммунального хозяйства в Украине / М. А. Мызникова // Економіка і управління. - 2015. - № 4. - С. 35-41. - Режим доступу: http://nbuv.gov.ua/UJRN/econupr_2015_4_8
10. Насонкина Н.Г. Анализ повреждаемости водопроводных и канализационных сетей / Н.Г. Насонкина, В.Н. Сахновская // Вестник луганского национального университета имени Владимира Даля. – 2019. - № 10 (28). – С. 122 - 130
11. Орлов В.А. Стратегия восстановления водопроводных и водоотводящих сетей. – Москва: Ассоциация строительных вузов, 2001. – 95 с.

12. Пастухов Д.Ю. Инновационные технологии в трубопроводной коммунальной системе городов / Д. Ю. Пастухов // TRANSPORT BUSINESS IN RUSSIA. – 2011. - № 11 - С. 12 – 14
13. Рыльцева Ю.А Системы визуального обследования и ремонта водопроводных и водоотводящих трубопроводов / Рыльцева Ю.А., Орлов В.А. // Инженерный вестник Дона, - 2020. - №4.
14. Сеничев А. В., Новикова А. И., Васильев П. В. Сравнение глубокого обучения с традиционными методами компьютерного зрения в задачах идентификации дефектов // Молодой исследователь Дона, - 2020, - №4. - №25.
15. Скибо Д. В. Методы теледиагностики как способ повышения надёжности водоотводящих сетей / Д.В. Скибо, Н.Д. Пельменёва Н. Д. // Журнал «Молодежный вестник ИРГТУ». – 2016. - № 4 – С. 33 - 37 <https://www.elibrary.ru/item.asp?id=27516093>
16. Спесивец Р.В., Орехова В.И. Особенности автоматизации систем водоснабжения и водоотведения/В сборнике: Вестник научно-технического творчества молодежи Кубанского ГАУ. В 4-х томах. Составители А. Я. Барчукова, Я. К. Тосунов; под редакцией А. И. Трубилина, ответственный редактор А. Г. Кощаев. 2016. С. 49- 53
17. Храменков С.В. Оптимизация восстановления водоотводящих сетей. – Москва: Стройиздат. - 2002. – 185с.
18. Храменков С.В. Стратегия модернизации водопроводной сети. – Москва: Стройиздат, 2005. – 398 с.
19. Ministry for Regional Development, Building and Housing of Ukraine (2017), "National report on drinking water quality and drinking water supply in Ukraine in 2016", available at: <http://www.minregion.gov.ua/napryamkidiyalnosti/zkhk/teplo-vodopostachannya-ta-vodovidvedennya/natsionalna-dopovid/proekt-natsionalnoyidopovidi-pro-yakist-pitnoyi-vodi-ta-stan-pitnogovodopostachannya-v-ukrayini-u-2016-rotsi/> (Accessed 10 Nov 2018).
20. Accounting Chamber of Ukraine (2016), "On the results of the audit of the implementation of the tasks and measures envisaged by the National target program "Drinking water of Ukraine" for 2011-2020", available at: http://www.ac-rada.gov.ua/doccatalog/document/16748257/R_RP_2-1.pdf?subportal=main (Accessed 10 Nov 2018)
21. Andy Lee. Comparing Deep Neural Networks and Traditional Vision Algorithms in Mobile Robotics / Niall O' Mahony. — Conference : Computer Vision Conference (CVC) 2019At: Las Vegas. 2019. 9 p.

22. Arel, I., Rose, D. C., Karnowski, T. P. Deep machine learning – a new frontier in artificial intelligence research. *IEEE Computational Intelligence Magazine* 5(4), 13–18 (2010).
23. Data Science Team Лог-люсс.
<https://datascience.eu/ru/%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5/%D0%BB%D0%BE%D0%B3-%D0%BB%D0%BE%D1%81%D1%81/>
24. Suzanna B., Geoffrey H., “A self-organizing neural network that discovers surfaces in random-dot stereograms”, *Nature*, vol. 355, pp. 161-163, 1992.
25. Haurum J., Bahnsen B., Pedersen C. H., Moeslund T. B., “Water Level Estimation in Sewer Pipes Using Deep Convolutional Neural Networks”, *Water*, vol. 12, no. 3412, pp. 2-14, 2020, doi:10.3390/w12123412
26. Haurum J.B., Moeslund T. B., “A survey on image-based automation of CCTV and SSET sewer inspections”, *Automation in Construction*, vol. 111, no. 103061 pp. , 2020. doi.org/10.1016/j.autcon.2019.103061
27. L'vivvodokanal (2018), "Project of reconstruction of Bortnitskaya aeration station", available at: http://ecoforum-lviv.com.ua/wp-content/uploads/2018/10/3_5-Lysiuk-Oleh.pdf (Accessed 10 Nov 2018)
28. Sandler M., Howard A., Zhu M., Zhmoginov A., Chen L.-C., “MobileNetV2: Inverted Residuals and Linear Bottlenecks”, *IEEE/CVF, Conference on Computer Vision and Pattern Recognition*. Salt Lake City, UT, USA: IEEE, 2018. pp. 4510-4516. doi: 10.1109/CVPR.2018.00474
29. NASSCO, Pipeline Assessment and Certification Program (PACP) v.7.0.3, 2018
30. Moradi S., Zayed T., Golkhoo F., “Review on Computer Aided Sewer Pipeline Defect Detection and Condition Assessment” *Infrastructures*, vol.4, no.10, pp. 1-13, 2019.
31. Verkhovna Rada of Ukraine (2005), The Law of Ukraine "On the National State Program called "Potable Water of Ukraine" for 2006—2020", available at: <http://zakon.rada.gov.ua/laws/show/2455-15> (Accessed 10 Nov 2018).

32. Yin X., Chen Y., A. Bouferguene “A deep learning-based framework for an automated defect detection system for sewer pipes”, *Automation in Construction*, vol. 109, no. 102967, pp. 1-16, 2020.
33. Bengio Y., Monperrus M., “Non-Local Manifold Tangent Learning”, *Neural Information and Processing Systems*. Downtown Branch, Montreal, pp.129-136, 2004.
34. Kaiming H., Xiangyu Z., Shaoqing R., Jian S., “Deep Residual Learning for Image Recognition”, *Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA: IEEE, 2016. pp. 778.
35. Zhao Z., Zheng P., Xu S., Wu X., “Object Detection With Deep Learning: A Review”, *IEEE*, vol. 30, no. 11, 2019, doi: 10.1109/TNNLS.2018.2876865.
36. Khan A., Sohail A., Zahoora U., Qureshi A. S., “A Survey, of the Recent Architectures of Deep Convolutional Neural Networks”, 2019, pp. 1-62. doi: 10.1007/s10462-020-09825-6.
37. Chen X., Gong C., Ma C., Huang X., Yang J., “Privileged Semi-Supervised Learning”, *IEEE*, 2018, doi:10.1109/icip.2018.8451098
38. Collobert R., Weston J., Bottou L., Karlen M., Kavukcuoglu K., Kuksa P., “Natural language processing (almost) from scratch”, *Journal of Machine Learning Research*, vol.12, pp. 2493–2537, 2011.
39. Davide C., *Siamese neural networks: an overview*, *Artificial Neural Networks*. New York City, 2020, pp. 73-94.
40. Bach F., Jenatton R., Mairal J., Obozinski G., *Structured sparsity through convex optimization*, *CoRR*, 2011.
41. Hinton G. Krizhevsky A., Wang S., *Transforming autoencoders*, *ICANN*, 2011.
42. Mobahi H., Collobert R., Weston J., *Deep learning from temporal coherence in video*, *ICML*, 2009.
43. Wiskott L., Sejnowski T., “Slow feature analysis: Unsupervised learning of invariances”, *Neural Computation*, vol. 14, pp. 715-770, 2002.
44. Goodfellow I., Bengio Y., Courville A., *Deep Learning*, The MIT Press. 2016, pp. 326-366. URL: <https://www.deeplearningbook.org/>
45. Girshick R., Donahue J., Darrell T. and J. Malik "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, pp. 580–587.
46. Lazzeri F., Глубокое обучение, машинное обучение и искусственный интеллект. - 08.09.2021. <https://docs.microsoft.com/ru-ru/azure/machine-learning/concept-deep-learning-vs-machine-learning>

Додаток А

```

import os

import shutil

shutil.rmtree('%content/dataset_original/to_224_padding/train/none')
shutil.rmtree('%content/dataset_original/to_224_padding/test/none')

if os.path.exists('%content/dataset_original/to_224_padding/train/none'):
    os.rmdir('%content/dataset_original/to_224_padding/train/none')
if os.path.exists('%content/dataset_original/to_224_padding/test/none'):
    os.rmdir('%content/dataset_original/to_224_padding/test/none')

train_dir, test_dir =
('%content/dataset_original/to_224_padding/train', '%content/dataset_original/to_224_padding/test'
)

import tensorflow as tf

from tensorflow.keras.models import Model, Sequential, load_model

from tensorflow.keras.layers import Dense, GlobalAveragePooling2D, Conv2D, MaxPooling2D,
Activation, Dropout, Flatten, Dense, GlobalMaxPooling2D

from tensorflow.keras.optimizers import Adam, RMSprop

from tensorflow.keras.metrics import categorical_crossentropy

from tensorflow.keras.preprocessing import image

from tensorflow.keras.preprocessing.image import ImageDataGenerator, array_to_img,
img_to_array, load_img

from tensorflow.keras.applications.efficientnet import preprocess_input

from tensorflow.keras.callbacks import TensorBoard, EarlyStopping, ModelCheckpoint

import neural_structured_learning as nsl

```

```
import numpy as np
from IPython.display import Image

from sklearn.utils import class_weight

from tensorflow.keras.regularizers import L1, L2

import pandas as pd
import os
import shutil

IMAGE_SIZE = 224
TRAINING_DIR = train_dir
VAL_DIR = test_dir
BATCH_SIZE = 64

adv_config = nsl.configs.make_adv_reg_config(
    multiplier=0.2,
    adv_step_size=0.2,
    adv_grad_norm='infinity'
)

def normalize(features):
    features[IMAGE_INPUT_NAME] = tf.cast(
        features[IMAGE_INPUT_NAME], dtype=tf.float32) / 255.0
    return features

def convert_to_tuples(features):
    return features[IMAGE_INPUT_NAME], features[LABEL_INPUT_NAME]
```

```

def convert_to_dictionaries(image, label):
    return {IMAGE_INPUT_NAME: image, LABEL_INPUT_NAME: label}

base_adv_model = build_base_model(HPARAMS)
adv_model = nsl.keras.AdversarialRegularization(
    base_adv_model,
    label_keys=[LABEL_INPUT_NAME],
    adv_config=adv_config
)

train_set_for_adv_model = train_dataset.map(convert_to_dictionaries)
test_set_for_adv_model = test_dataset.map(convert_to_dictionaries)

adv_model.compile(optimizer='adam', loss='sparse_categorical_crossentropy',
                  metrics=['acc'])
adv_model.fit(train_set_for_adv_model, epochs=HPARAMS.epochs)

import cv2
from tensorflow.python.keras.applications.mobilenet_v2 import preprocess_input

data_generator = ImageDataGenerator(
    preprocessing_function = preprocess_input,
    # rotation_range=90,
    width_shift_range=0.2,
    height_shift_range=0.2,
    # rescale= 1. / 255,
    shear_range=0.2,
    zoom_range=0.2,
)

train_generator = data_generator.flow_from_directory(TRAINING_DIR,

```

```

        target_size=(IMAGE_SIZE, IMAGE_SIZE),
        shuffle=True,
        seed=13,
        class_mode='categorical',
        batch_size=BATCH_SIZE)

valid_generator = data_generator.flow_from_directory(VAL_DIR,
        target_size=(IMAGE_SIZE, IMAGE_SIZE),
        shuffle=False, #True,
        seed=13,
        class_mode='categorical',
        batch_size=BATCH_SIZE,
        # subset="validation"
    )

class_weights = class_weight.compute_class_weight(
    class_weight='balanced',
    classes=np.unique(train_generator.classes),
    y=train_generator.classes)
print(class_weights)

class_number = len(os.listdir(TRAINING_DIR))

labels = train_generator.class_indices
labels = dict([(value, key) for key, value in labels.items()])
print(class_number)
print(labels)

def create_mobile_net():
    model = tf.keras.applications.mobilenet.MobileNetV2(
        input_shape=(224, 224, 3), alpha=0.25,

```



```

        include_top=True, weights=None, classes=class_number, classifier_activation='softmax'
    )

    return model

def get_encoder_basic(image_size=IMAGE_SIZE, width=128):
    extractor = tf.keras.applications.MobileNetV2(
        input_shape=(image_size, image_size, 3),
        alpha=0.25,
        include_top=False,
        weights='imagenet' )
    return extractor

def create_model_feature_extraction():
    encoder = get_encoder_basic()

    model = Sequential([
        encoder,
        Flatten(),
        Dense(512, activation='relu'),
        Dropout(0.25),
        Dense(class_number, activation='softmax')
    ])
    return model

def create_model_feature_extraction_2(shape: int = 224):
    inputs = tf.keras.Input(shape=(shape, shape, 3))
    base_model = get_encoder_basic()

    x = base_model(inputs)
    x = tf.keras.layers.GlobalAveragePooling2D()(x)

```

```

# x = tf.keras.layers.Dropout(0.2)(x)
x = tf.keras.layers.Dense(class_number * 2, kernel_regularizer=tf.keras.regularizers.L2(1e-4),
                           bias_regularizer=tf.keras.regularizers.L2(1e-4),
                           activity_regularizer=tf.keras.regularizers.L2(1e-4),
                           activation=tf.keras.activations.softmax)(x)
x = tf.keras.layers.Dropout(0.3)(x)
x = tf.keras.layers.Flatten()(x)
x = tf.keras.layers.Dense(class_number, kernel_regularizer=tf.keras.regularizers.L2(1e-4),
                           bias_regularizer=tf.keras.regularizers.L2(1e-4),
                           activity_regularizer=tf.keras.regularizers.L2(1e-4),
                           activation=tf.keras.activations.softmax)(x)

return tf.keras.Model(inputs, x)

def get_base_model(shape: int = 224) -> tf.keras.Model:
    model = tf.keras.applications.MobileNetV2(
        input_shape=(shape, shape, 3),
        alpha=0.35,
        include_top=False,
        weights='imagenet',
        classes=class_number,
    )
    # model.trainable = True
    # for layer in model.layers[:100]:
    #     layer.trainable = False
    model.trainable = False

    return model

def create_model_transfer_learning(shape: int = 224):
    inputs = tf.keras.Input(shape=(shape, shape, 3))

```

```

base_model = get_base_model()

x = base_model(inputs)
x = tf.keras.layers.GlobalAveragePooling2D()(x)
x = tf.keras.layers.Dropout(0.2)(x)
x = tf.keras.layers.Dense(class_number * 3, kernel_regularizer=tf.keras.regularizers.L2(1e-4),
                           bias_regularizer=tf.keras.regularizers.L2(1e-4),
                           activity_regularizer=tf.keras.regularizers.L2(1e-4),
                           activation=tf.keras.activations.softmax)(x)
x = tf.keras.layers.Dropout(0.5)(x)
x = tf.keras.layers.Dense(class_number * 2, kernel_regularizer=tf.keras.regularizers.L2(1e-4),
                           bias_regularizer=tf.keras.regularizers.L2(1e-4),
                           activity_regularizer=tf.keras.regularizers.L2(1e-4),
                           activation=tf.keras.activations.softmax)(x)
x = tf.keras.layers.Dropout(0.3)(x)
x = tf.keras.layers.Flatten()(x)
x = tf.keras.layers.Dense(class_number, kernel_regularizer=tf.keras.regularizers.L2(1e-4),
                           bias_regularizer=tf.keras.regularizers.L2(1e-4),
                           activity_regularizer=tf.keras.regularizers.L2(1e-4),
                           activation=tf.keras.activations.softmax)(x)

return tf.keras.Model(inputs, x)

def get_base_model(shape: int = 224) -> tf.keras.Model:
    model = tf.keras.applications.MobileNetV2(
        input_shape=(shape, shape, 3),
        alpha=0.35,
        include_top=False,
        weights='imagenet',
        classes=class_number,
    )
    # model.trainable = True

```

```

# for layer in model.layers[:100]:
#     layer.trainable = False
model.trainable = False

return model

def create_model_transfer_learning_2(shape: int = 224):
    inputs = tf.keras.Input(shape=(shape, shape, 3))
    base_model = get_base_model()

    x = base_model(inputs)
    x = tf.keras.layers.GlobalAveragePooling2D()(x)
    x = tf.keras.layers.Dropout(0.2)(x)
    x = tf.keras.layers.Flatten()(x)
    x = tf.keras.layers.Dense(class_number, kernel_regularizer=tf.keras.regularizers.L2(1e-4),
                               bias_regularizer=tf.keras.regularizers.L2(1e-4),
                               activity_regularizer=tf.keras.regularizers.L2(1e-4),
                               activation=tf.keras.activations.softmax)(x)

    return tf.keras.Model(inputs, x)

#                                     modelLoad_weights("%content/drive/MyDrive/Colab
Notebooks/PIPE/PipeWaterLevel/checkpoints/checkpoint-23-0.7074.hdf5")

#tf.saved_model.save(model,                                     "%content/drive/MyDrive/Colab
Notebooks/PIPE/PipeWaterLevel/waterlevel_model_3102021/")

def macro_f1(y, y_hat, thresh=0.5):
    """Compute the macro F1-score on a batch of observations (average F1 across labels)

    Args:
        y (int32 Tensor): labels array of shape (BATCH_SIZE, N_LABELS)

```

`y_hat` (float32 Tensor): probability matrix from forward propagation of shape (BATCH_SIZE, N_LABELS)

`thresh`: probability value above which we predict positive

```

run=0
run = run + 1 if 'run' in locals() else 1
LOG_DIR = "%content/drive/MyDrive/Colab Notebooks/PIPE/PipeWaterLevel/checkpoints"
LOG_FILE_PATH = LOG_DIR + '%checkpoint-[epoch:02d]-{val_loss:4f}.hdf5'
tensorboard = TensorBoard(log_dir=LOG_DIR, write_images=True)
reduce_lr = tf.keras.callbacks.ReduceLROnPlateau(monitor='val_loss', factor=0.1, patience=1,
min_lr=1e-9)
checkpoint = ModelCheckpoint(filepath=LOG_FILE_PATH, monitor='val_loss', verbose=1,
save_best_only=True)
early_stopping = EarlyStopping(monitor='val_loss', patience=8, verbose=1)

test_steps_per_epoch = np.math.ceil(valid_generator.samples / valid_generator.batch_size)
train_steps_per_epoch = np.math.ceil(train_generator.samples / train_generator.batch_size)

print("test_steps_per_epoch,   train_steps_per_epoch   =   ", (test_steps_per_epoch,
train_steps_per_epoch))

print(checkpoint)
print(len(class_weights))
history = model.fit_generator(generator=train_generator,
                             class_weight={idc for id, c in enumerate(class_weights)},
                             steps_per_epoch=train_steps_per_epoch,
                             validation_data=valid_generator,
                             validation_steps=test_steps_per_epoch,
                             epochs=EPOCHS,
                             verbose=1,
                             callbacks=[tensorboard, checkpoint, early_stopping, reduce_lr])

```

```

from numpy import argmax
from sklearn.metrics import confusion_matrix
from matplotlib import pyplot as plt
import seaborn as sn
import pandas as pd
y_pred = []
y_true = []
import cv2
from tensorflow.python.keras.applications.mobilenet import preprocess_input

def preprocess_input_graph(img):
    img_batch_tensor = tf.convert_to_tensor(np.array([img]), dtype=tf.float32)
    img_batch_tensor = img_batch_tensor[None, :-1]
    img_batch_tensor = tf.reverse(img_batch_tensor, axis=[-1])
    return preprocess_input(img_batch_tensor)

labels_list = ['0', '10', '20', '30', '40', '5', '50', '60']#os.listdir(VAL_DIR)
print(labels_list)
for y_true_id, y_true_name in enumerate(labels_list):
    print("START " + y_true_name)
    for img_file in os.listdir(os.path.join(VAL_DIR, y_true_name)):

        img = cv2.imread(os.path.join(VAL_DIR, y_true_name, img_file))
        if img is None:
            continue

        batch = preprocess_input_graph(img)

        res = model.predict(batch)
        id_pred = np.argmax(res[0])
        y_pred.append(id_pred)

```

```
    y_true.append(y_true_id)
#

cf_m = confusion_matrix(y_true, y_pred, normalize=True)
df_cm = pd.DataFrame(cf_m, index=labels_list,
                    columns=labels_list)
plt.figure(figsize=(15, 10))
sns.heatmap(df_cm, annot=True, cmap='Blues')
plt.show()
```