

ІНФОРМАТИКА, ОБЧИСЛЮВАЛЬНА ТЕХНІКА ТА АВТОМАТИЗАЦІЯ

УДК 004.896

DOI <https://doi.org/10.32782/2663-5941/2022.6/12>

Амбросьонюк А.А.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Сазонов А.Ю.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Черепанська І.Ю.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

Лукінюк М.В.

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

СИСТЕМА ДЕТЕКТУВАННЯ ПОСТРАЖДАЛИХ В РЕЗУЛЬТАТІ ВИКИДІВ ХІМІЧНО-АКТИВНИХ РЕЧОВИН

Технологічні процеси харчової, фармацевтичної, хімічної та інших галузей промисловості завжди супроводжуються ризиками виникнення аварійних ситуацій, які можуть бути викликані людськими або техногенними факторами, що спричиняють викиди хімічно- та біологічно-активних речовин, а також радіаційне забруднення довкілля. У разі виникнення хімічної аварії на виробництві життя чи смерть людини визначається в перші хвилини після інциденту, тому важливим є вживання необхідних заходів для допомоги постраждалим працівникам виробництва, зокрема проведення їх знезараження у перші хвилини аварії. Актуальним є також проведення ідентифікації та підрахунку кількості постраждалих у результаті аварії, що дозволить зменшити час на підготовку до знезараження та підвищити ефективність лікування за рахунок завчасного визначення необхідної кількості ліжок-місць у лікарнях залежно від кількості потерпілих. Метою статті є розроблення ефективного інструменту детектування людей та підрахунку їх на зображеннях з камер відеонагляду як основи процедури керування потерпілими в умовах викидів хімічно активних забруднювальних речовин. Для досягнення зазначеної мети використовується підхід перенесеного навчання (transfer learning), що полягає у застосуванні знань, отриманих під час виконання розв'язання однієї задачі, для іншої цільової задачі. Для навчання обрана згортова нейронна мережа InceptionResNetV2, яка є більш новою та потужнішою ніж більш ранні моделі. Згорткові мережі добре справляються з величезними наборами даних та ефективно навчаються на графічних процесорах за рахунок паралельних обчислень. Набір даних для тренування складався з зображень із камер відеонагляду, а також з файлу, де була порахована кількість людей на кожному фото. Після налаштування та тренування модель показала гарні результати, які можна побачити на графіках і зображеннях, тому вона може бути використаною в подальших дослідженнях.

Ключові слова: хімічні виробництва, аварійні ситуації, нейронна мережа, детектування людей, згортові нейронні мережі, екстренні служби.

Постановка проблеми. Технологічні процеси харчової, фармацевтичної, хімічної та інших галузей промисловості завжди супроводжуються ризи-

ками виникнення аварійних ситуацій, що можуть бути викликані людськими або техногенними факторами, які спричиняють викиди хімічно- та

біологічно-активних речовин, а також радіаційне забруднення довкілля. Крім того, в умовах сьогоднішнього дня, коли велика кількість підприємств перебувають безпосередньо у зоні бойових дій, ризики забруднення довкілля та, відповідно, зараження інженерно-технічного персоналу підприємств є надзвичайно високими [1].

У разі виникнення хімічної аварії на виробництві життя чи смерть людини визначається в перші хвилини після інциденту. Доведено [2], що під час масштабних інцидентів найбільш поширеною причиною смерті є те, що постраждалим вчасно не було надано кваліфіковану медичну допомогу. Тому дуже важливим є вживання необхідних заходів для допомоги постраждалим працівникам виробництва та проведення їх знезараження у перші хвилини аварії. Не менш актуальним є завдання ідентифікації та підрахунку кількості постраждалих у результаті аварії, що дозволить зменшити час на підготовку до знезараження та підвищити ефективність лікування за рахунок завчасного визначення необхідної кількості ліжок-місць у лікарнях в залежності від кількості потерпілих. Після отримання невідкладної допомоги постраждалий повинен пройти через станцію знезараження, перш ніж йому буде надано більш серйозну допомогу в чистому середовищі. Ця процедура займає 10–20 хвилин. Процедуру знезараження у деяких випадках необхідно проводити поблизу активної зони забруднення або на території лікарні, щоб доставляти постраждалих безпосередньо до медичних установ без попереднього проходження знезараження [1].

Підрахунок постраждалих допоможе швидко зрозуміти обсяги проблеми, щоб негайно вжити необхідні заходи й ефективно спланувати процедуру знезараження та лікування.

Підрахунок кількості об'єктів є складним завданням, особливо у разі використання систем відеоспостереження, оскільки таку процедуру треба застосовувати до послідовності кадрів, а не до одного статичного зображення. Це можливо реалізувати із використанням алгоритмів машинного навчання, де вхідними даними є дискретизована послідовність кадрів, а виходом – дискретне значення кількості об'єктів.

Аналіз останніх досліджень і публікацій.

Враховуючи підхід із використанням алгоритмів машинного навчання, перш за все необхідно визначитись із архітектурою нейронної мережі, що забезпечить якісне та швидке виконання поставленого завдання. Для цього можна застосувати різні підходи, один із яких полягає у тому, що

розглядають кількість елементів на зображенні як деяку мітку та перетворюють її на певну задачу класифікації або використовують цілковито згорткову архітектуру, де кінцевий згортковий вихід може розглядати номери об'єктів у області інтересу, а потім виконувати підсумування їх кількості. Найпоширеніший тип нейронних мереж, які використовуються на сьогодні для роботи із різними видами зображень, – згорткові [3], що є близьким аналогом до оброблення зображення людиною за допомогою рецептивних полів, при цьому окремі нейрони сітківки одержують стимули з різних областей поля зору. Інформація від кількох нейронів сітківки ока згодом передається нейронам далі за ланцюжком. Згорткові мережі схожі на інші, але мають додатковий рівень складності через те, що вони побудовані із послідовності згорткових шарів. Згорткові мережі добре справляються з величезними наборами даних та ефективно навчаються на графічних процесорах за рахунок паралельних обчислень.

Найефективнішими архітектурами згорткових мереж сьогодні є ResNet50 та InceptionResNetV2.

ResNet50 [4] являє собою згорткову нейромережу глибиною у 50 шарів, 48 із яких є згортковими, один шар максимізаційного агрегування (max pooling) і один – усереднювального агрегування (average pooling). Початковою архітектурою ResNet була ResNet-34, що складалася з 34 зв'язаних шарів. Це дало можливість додати більше згорткових шарів до мережі, не стикаючись із проблемою зникаючого градієнта, використовуючи концепцію коротких з'єднань. Таке з'єднання пропускає деякі шари, перетворюючи звичайну мережу на залишкову. Архітектура ResNet дотримується двох основних правил проектування. По-перше, кількість фільтрів у кожному шарі є однаковою та залежить від розміру вихідної карти ознак. По-друге, якщо розмір картки ознак зменшити вдвічі, кількість фільтрів подвоїться, щоб зберегти тимчасову складність кожного шару. ResNet50 використовує конструкцію вузького місця стандартного блоку. Залишковий блок із вузьким місцем використовує згортки 1×1 , відомі як вузьке місце, які зменшують кількість параметрів та матричних множень. Це дозволяє підвищити швидкість навчання кожного шару.

Іншою архітектурою є InceptionResNetV2, яка також є згортковою та попередньо навченою на понад мільйони зображень із бази даних ImageNet. InceptionResNetV2 складається зі 164 шарів і може класифікувати зображення за тисячами категорій об'єктів, таких як людина, олівець і багато тва-

рин. Набір має вхідний розмір зображення 299 на 299 пікселів, а вихід являє собою передбачуваний список ймовірних класів. Inception-ResNet поєднує в собі архітектуру Inception із залишковими сполуками [5].

Архітектуру [6] InceptionResNetV2 схематично представлено на рис. 1.

Використання залишкових зв'язків покращує швидкість навчання, що є вагомим аргументом для їх використання. За кожним початковим блоком надходить шар розширення фільтра, який використовується для збільшення розмірності банку фільтрів перед додаванням, щоб відповідати глибині введення. Це потрібно, щоб компенсувати зменшення розмірності, викликане початковим блоком [5].

За допомогою зменшення залишків перед додаванням їх до активації попереднього шару вдалося стабілізувати тренування. Для масштабування залишків коефіцієнти масштабування мають бути від 0,1 до 0,3.

Перевагою InceptionResNetV2 є те, що ця модель є більш новою та потужнішою, вона має більшу кількість шарів, що підвищує якість і точність результату. Цю модель дуже легко налаштувати для нової задачі, зберігаючи корисні знання оригінального класифікатора, і результат буде отриманий набагато швидше. Недоліком моделі є те, що, незважаючи на високу точність, модель не є ідеальною, і можуть бути виникати похибки.

Формулювання цілей статті. Метою цієї статті є розроблення ефективного інструменту детектування людей та підрахунку їх на зображенні як основи процедури керування потерпілими в умовах викидів хімічно активних забруднювальних речовин. Для досягнення зазначеної мети вико-

ристовується підхід перенесеного навчання (transfer learning), що полягає у застосуванні знань, отриманих під час розв'язання однієї задачі, до іншої цільової задачі.

Для цього буде обрано згорткову нейронну мережу InceptionResNetV2, переваги якої було розглянуто вище. Модель буде завантажено з додатків Keras з оригінальними вагами. Щоб модель впоралася з поставленою задачею, вона буде натренована. Для тренування було обрано набір даних з платформи Kaggle. Набір складається з зображень у форматах NumPy та JPG, а також файла з порахованою кількістю осіб на кожному кадрі. Швидкість навчання буде коригуватись щоразу, коли втрати під час перевірки будуть погіршуватись.

Виклад основного матеріалу. Перед тим як почати навчання, потрібно проаналізувати вхідні дані та завантажити їх в програму. Зображення мають розмір 480×640 пікселів у 3 каналах однієї точки, записані веб-камерою в торговому центрі, але на кожному кадрі зображено різну кількість людей, що є проблемою підрахунку натовпу. Для кожного зображення є цифрове значення з кількістю людей на ньому рис. 2.

На рис. 3 можна побачити, з якою частотою змінюється кількість людей на кадрах. Цільові значення (кількість людей) коливаються від 13 до 53 із середнім значенням 31,16. Значення зазвичай розподіляються з середнім значенням, близьким до середнього. Загальна кількість зображень 2000.

Спочатку необхідно отримати зображення для того, щоб потім на їхній основі сформувати набори даних для тренування та перевірки моделі. Для цього з файла було завантажено зображення і з зазначенням кількості людей на ньому та без.

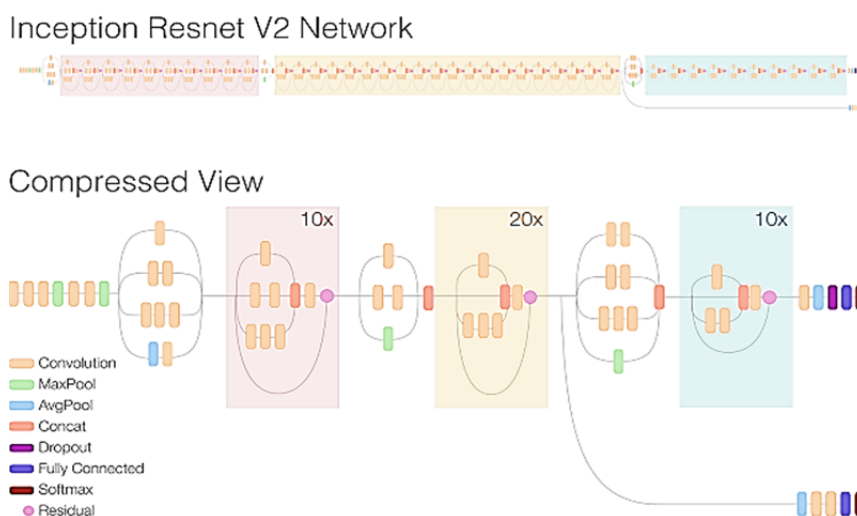


Рис. 1. Архітектура InceptionResNetV2

	id	count	path
0	1	35	frames/frames/seq_000001.jpg
1	2	41	frames/frames/seq_000002.jpg
2	3	41	frames/frames/seq_000003.jpg
3	4	44	frames/frames/seq_000004.jpg
4	5	41	frames/frames/seq_000005.jpg

Рис. 2. Назва зображення і кількість людей на ньому

Для тренування було обрано 85 % даних, тобто 1700 зображень. Для перевірки обрано решту даних, тобто 15 %. Далі створюються набори даних для тренування та перевірки і перетворюються у набір даних із тензорами зображень і мітками.

Після отримання наборів даних починається створення моделі. Функціональну модель було завантажено з додатків Keras за допомогою функції `tf.keras.applications.InceptionResNetV2`, яка повертає модель класифікації зображень Keras, що опціонально завантажується з вагами попередньо навченими в ImageNet. Для функції можна вказати, як будуть завантажені попередньо навчені ваги. Можна обрати випадкову ініціалізацію, попереднє навчання на ImageNet або шлях до файла ваг. Було обрано випадкову ініціалізацію і встановлено `pooling = 'avg'` – це означає, що глобальне середнє об'єднання буде застосовано до виходу останнього згорткового блоку *i*, таким чином, виходом моделі буде 2D-тензор.

Після цього необхідно згрупувати лінійний стек шарів за допомогою функції `Sequential`. Вона надає функції навчання та виведення для цієї моделі, допомагає сформувати кластер шару, який лінійно укладається у `tf.keras.Model`. На вхід зазначеній функції передано розміри зображення, функціональну модель та щільно пов'язаний шар.

Далі йде налаштування моделі, для чого необхідно використати метод `compile` в якому на вході можна вказати потрібний оптимізатор. Визначення оптимізатора ітеративно покращує ваги та швидкість навчання, щоб мінімізувати втрати. Алгоритм оптимізації Адама є розширенням стохастичного градієнтного спуску, який останнім часом набув ширшого поширення широкого розповсюдження у додатках глибокого навчання комп'ютерному зору та обробці природної мови. Алгоритм використовується замість класичної процедури стохастичного градієнтного спуску

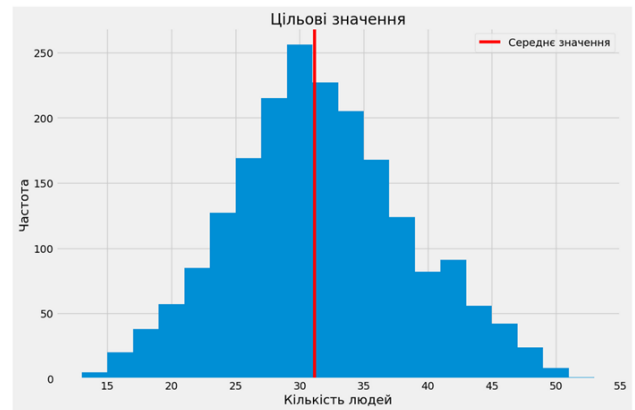


Рис. 3. Цільові значення (кількість людей) на зображеннях

для ітеративного оновлення ваги мережі на основі навчальних даних. Переваги алгоритму Адама в тому, що він простий у реалізації, потребує невеликого обсягу пам'яті й є ефективним у обчислювальному відношенні. Стохастичний градієнтний спуск підтримує єдину швидкість навчання для всіх оновлень ваги і ця швидкість під час навчання не змінюється. Швидкість навчання підтримується для кожної ваги мережі й адаптується окремо з розвитком навчання.

Обраний алгоритм оптимізації передається на вхід функції `Sequential`. Важливим параметром для оптимізатора є швидкість навчання. Цей параметр контролює кількість оновлення вагових коефіцієнтів під час кожного раунду навчання. Швидкість навчання вказано 0,001.

Також для методу `compile` вказується `loss` та `metrics`, тобто функції втрат та список метрик. Мета функцій втрат полягає у тому, щоб обчислити кількість ітерацій та звести їх до мінімуму під час тренування. За функцію втрат обрано `MeanSquaredError (MSE)`. Вона обчислює середній квадрат помилок між мітками та прогнозами. MSE є показником якості оцінювача, оскільки вона обчислюється з квадрата Евклідової відстані, цей показник завжди має позитивне значення, а помилка зменшується в міру наближення його до нуля. Параметр `metrics` – це список метрик, які будуть оцінюватися моделлю під час навчання та тестування. За метрику вказано обрано `MeanAbsoluteError`, яка обчислює середнє значення абсолютної різниці між мітками та прогнозами.

Після налаштування моделі починається її тренування. Для цього буде використовуватись метод `fit`, який навчає модель протягом фіксованої кількості епох (ітерацій у наборі даних). На вхід методу передається набір даних для тренування

та набір даних для перевірки, які було сформовано вище. Також важливим параметром є кількість епох для навчання моделі, яка буде дорівнювати 500. Для швидкості й якості результатів задається список зворотних дзвінків для застосування під час навчання. Встановлено відпалювач швидкості навчання ReduceLRonPlateau, щоб швидкість навчання зменшувалася під час навчання для того, аби ефективно досягти глобального мінімуму функції втрат. LR динамічно зменшується, якщо оцінка не покращується. Це зберігає перевагу швидкого часу обчислення з високим LR на початку. Для моніторингу обрано середній квадрат помилок для набору перевірки. Кількість епох, у яких не спостерігається жодного покращання, встановлено як 1, а мінімальне значення швидкості як 0,000001.

Для забезпечення зворотних дзвінків встановлено EarlyStopping, яке зупиняє навчання, коли показник, що відстежується, перестає покращу-

ватися. За монітор також обрано середній квадрат помилок для набору перевірки. Кількість епох без покращення – після чого навчання буде зупинено – встановлено таким, що дорівнює 10.

Отже після всієї підготовки можна приступати до навчання, яке займає певний проміжок часу. Процес навчання можна спостерігати на рис. 4.

Як можна побачити з рис. 4, LR динамічно зменшується, якщо оцінка не покращується. Значення середньої квадратичної похибки та середньої абсолютної похибки також зменшуються. Навчання закінчується на епісі 117, оскільки показники перестають покращуватися. Середня абсолютна похибка дорівнює 3,11. На рис. 5 можна ознайомитись з графіком, на якому показано, як змінювалась MAE і коли вона досягла найкращого значення.

Результат підрахунку також можна побачити на рис. 6, де показано різні зображення та підраховану кількість людей на них.

```
Epoch 1/500
107/107 [=====] - 1214s 11s/step - loss: 65.1584 - mean_absolute_error: 6.3754 - val_loss: 48.5355 - val_mean_absolute_error: 6.1197 - lr: 0.0010
Epoch 2/500
107/107 [=====] - ETA: 0s - loss: 45.7749 - mean_absolute_error: 5.4650
Epoch 2: ReduceLRonPlateau reducing learning rate to 0.0007500000356230885.
107/107 [=====] - 1429s 13s/step - loss: 45.7749 - mean_absolute_error: 5.4650 - val_loss: 50.2757 - val_mean_absolute_error: 6.2717 - lr: 0.0010
Epoch 3/500
107/107 [=====] - 1414s 13s/step - loss: 41.8126 - mean_absolute_error: 5.0426 - val_loss: 30.9484 - val_mean_absolute_error: 4.8720 - lr: 7.5000e-04
Epoch 4/500
107/107 [=====] - 1443s 13s/step - loss: 36.0991 - mean_absolute_error: 4.6904 - val_loss: 30.8461 - val_mean_absolute_error: 4.8715 - lr: 7.5000e-04
Epoch 5/500
107/107 [=====] - 1495s 14s/step - loss: 33.7839 - mean_absolute_error: 4.5527 - val_loss: 29.8709 - val_mean_absolute_error: 4.7915 - lr: 7.5000e-04
Epoch 6/500
107/107 [=====] - ETA: 0s - loss: 31.7482 - mean_absolute_error: 4.4216
```

Рис. 4. Процес навчання моделі

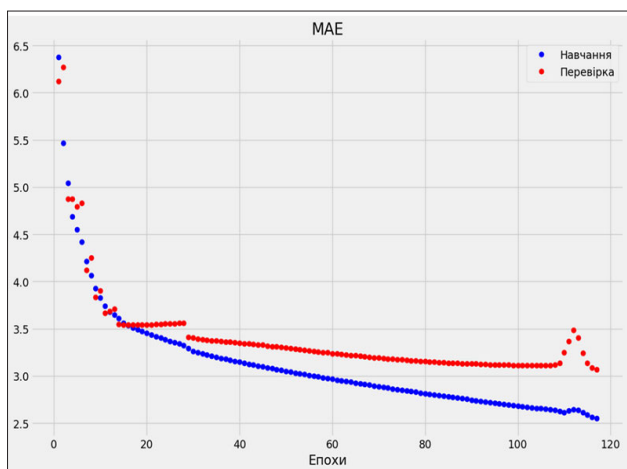


Рис. 5. Зміна значення середньої абсолютної похибки

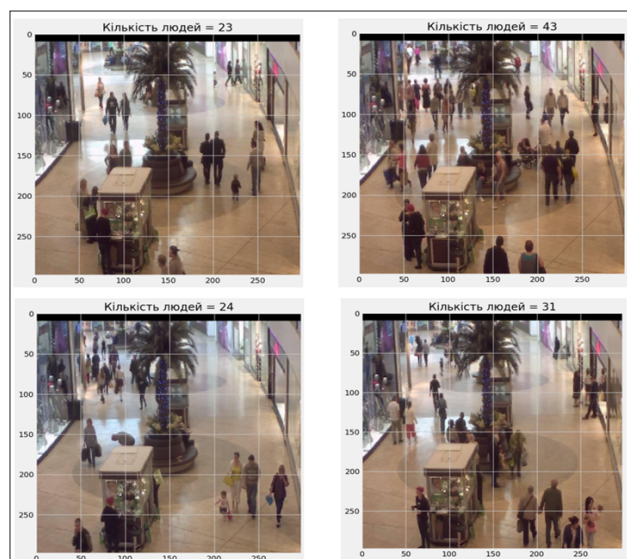


Рис. 6. Підрахунок людей за допомогою моделі

Висновки. В результаті проведеної роботи було обрано архітектуру нейронної мережі InceptionResNetV2, оскільки вона була попередньо навчена на наборах даних і є потужнішою за інші моделі. Після проведення тренування моделі було отримано непогані показники, які засвідчують, що модель швидко вчиться та показує більш високу точність ніж більш ранні моделі. На якість даних також впливають вхідні дані. Вхідними даними були фото з камер відео нагляду та файл з певною кількістю людей на кожному фото. Файл

з мітками був порохований вручну, тому треба ще враховувати людський фактор, оскільки можуть бути помилки під час аналізу такого об'єму даних людиною. Також набір даних був відносно невеликий, оскільки це лише 2000 зображень, для кращих результатів у навчанні треба використовувати більшу кількість даних. Ще одним, не менш важливим, фактором є якість фото: чим вона вище, то кращими будуть показники. Тому ця модель є цілком придатною для підрахунку людей і може бути використана у подальших дослідженнях.

Список літератури:

1. Aruna C. Ramesh Triage, monitoring, and treatment of mass casualty events involving chemical, biological, radiological, or nuclear agents. *Journal of Pharmacy and Bioallied Sciences*. 2010. 2(3): 239–247.
2. James E. Lukaszewski. Common Ingredients of Crisis Situations – Victim Management is Paramount. *America's Crisis Guru*. URL: <https://www.summary.com/magazine/common-ingredients-of-crisis-situations-victim-management-is-paramount/> (дата звернення 28.10.2022).
3. Cheng J. Convolutional neural networks. *PathologyOutlines.com website*. URL: <https://www.pathologyoutlines.com/topic/informaticconvnet.html>
4. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Su. Deep Residual learning for image recognition. *Computer vision and pattern recognition*. URL: <https://arxiv.org/pdf/1512.03385.pdf> (дата звернення 28.10.2022).
5. Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke. Deep Inception-v4, inception-res net and the impact of residual connections on learning. *Computer vision and pattern recognition*. URL: <https://arxiv.org/pdf/1602.07261.pdf> (дата звернення 28.10.2022).
6. Alex Alemi. Improving Inception and Image Classification in TensorFlow. URL: <https://ai.googleblog.com/2016/08/improving-inception-and-image.html> (дата звернення 28.10.2022).

Ambrosonok A.A., Sazonov A.Yu., Cherepanska I.Yu., Lukiniuk M.V. VICTIMS DETECTION SYSTEM IN CHEMICALLY CONTAMINATED AREA

Technological processes of the food, pharmaceutical, chemical and other types of industry are always accompanied by risks of emergency situations that can be caused by human or technogenic factors which cause emissions of chemically and biologically active substances, as well as radiation pollution of the environment. In the event of a chemical accident at work, the life or death of a person is determined in the first minutes after the incident, therefore it is important to take the necessary measures to help the injured production workers, to carry out their disinfection in the first minutes of the accident. The task of identifying and counting the number of victims as a result of the accident is urgent, which will reduce the time for preparation for decontamination and increase the effectiveness of treatment due to early determination of the required number of beds in hospitals depending on the number of victims. The purpose of this article is to develop an effective tool for detecting people and counting them in the image from video surveillance cameras as a basis for the procedure of managing victims in the conditions of emissions of chemically active pollutants. To achieve this goal, the transfer learning approach is used, which consists in applying the knowledge obtained from one task to another target task. The convolutional neural network InceptionResNetV2, which is newer and more powerful than earlier models, was chosen for training. Convolutional networks handle large data sets well and are trained efficiently on GPUs through parallel computation. The data set for training consisted of images from video surveillance cameras, as well as a file where the number of people in each photo was counted. After tuning and training, the model showed good results, which can be seen in the graphs and images, so it can be used in further research.

Key words: chemical production, emergency situations, neural network, detection of people, convolutional neural networks, first responders.