

# НЕЙРОННА МЕРЕЖА ДЛЯ ВИЯВЛЕННЯ АРТЕФАКТІВ НА КТ-ЗОБРАЖЕННЯХ

Шликов В.В., доц., д.т.н.

[v.shlykov@kpi.ua](mailto:v.shlykov@kpi.ua)

Воляник О.М., магістр

[volyanik.oleg@gmail.com](mailto:volyanik.oleg@gmail.com)

Кафедра біомедичної інженерії

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

м. Київ, Україна

*Реферат* – Обґрунтовано необхідність створення системи для виявлення артефактів на КТ-зображеннях, створено архітектуру нейронної мережі в програмному середовищі Python за допомогою нейромережевої бібліотеки Keras та перевірено її точність за допомогою тестового набору зображень. Показано, що потенційно, можливості отриманої системи можуть бути вдосконалені для інших задач біомедичної інженерії.

*Ключові слова* – комп'ютерна томографія, нейронна мережа, згортоква нейронна мережа, артефакти, Python, бібліотека Keras, 3D-шаблон протеза.

## I. ВСТУП

У сучасній хірургії набувають широке застосування цифрові технології, які засновані на використанні спеціальних програмно-апаратних систем, які отримали найменування Computer Aided Surgery (CAS). CAS-технології призначені для вдосконалення і автоматизації діагностики, передопераційного планування, тренінгу хірурга-оператора, а також безпосередньо хірургічного втручання і інтраопераційного супроводу [1].

Розвиток CAS-технологій пов'язаний з науковими досягненнями, серед яких слід відзначити сучасну діагностику (КТ, МРТ), віртуальне комп'ютерне моделювання і адитивні технології (3D-друк шаблонів), що використовуються для виготовлення протезів, допоміжних елементів та імплантатів. Отже, створені до теперішнього часу CAS-системи можна охарактеризувати як об'єднання в єдиний технологічний ланцюжок медичних апаратних засобів, спеціалізованих програмних комплексів і комп'ютерних програм.

Одним з інформативних методів неінвазивної діагностики, що дозволяє проводити дослідження широкого спектру об'єктів (як біологічних, так і промислових) з різним хімічним складом і можливістю побудови тривимірних моделей за

отриманими даними є комп'ютерна томографія. Комп'ютерні томографи (КТ) найбільш часто використовуються в медичній діагностиці та характеризуються відносно низькими експлуатаційними витратами при високій пропускній здатності. Основним недоліком цього інформаційного методу є наявність рентгенівського випромінювання [2].

В цілому якість КТ-зображення характеризується п'ятьма факторами: просторовим дозволом, контрастністю, шумом або просторовою однорідністю, лінійністю і наявністю артефактів. Поява артефактів може не просто знизити візуальну якість зображень, але в ряді клінічних випадків зробити їх абсолютно непридатними для медичної діагностики. Артефактами зображень в комп'ютерній томографії є будь-яка невідповідність між КТ-числами реконструйованого зображення та істинними коефіцієнтами ослаблення контрасту матеріала об'єкта.

Наприклад, у досліджуваній області досить часто присутні об'єкти з високою щільністю, які зроблені з металу, що викликає на зображенні поява артефактів у вигляді смуг.

У зв'язку з цим всі артефакти можна поділити на чотири групи відповідно до причин, що викликали їх появу:

- фізичні процеси, що визначають механізм збору даних;
- чинники, які пов'язані з пацієнтом;
- несправність медичної апаратури;
- технологія сканування.

Для ідентифікації та аналізу об'єктів і артефактів на медичних зображеннях в системах машинного зору використовується системи CAS на базі методів машинного навчання [6, 8, 9]. Найбільш поширеними методами машинного навчання для задач класифікації [13] є штучні нейронні мережі (artificial neural network) [7], логістична регресія [1], метод опорних векторів Support Vector Machine (SVM) [71] та метод Random Forest [14].

Проте описані алгоритми обробки зображень видають більше помилок під час прогнозування артефактів, аніж нейронна мережа, яка здатна узагальнювати дані та обробляти великі їх обсяги. Отже, побудова систем розпізнавання КТ-зображень або інформаційних систем на основі експертних знань (pattern-recognition) та правил виокремлення ознак (feature extraction), в основу яких покладено методи машинного навчання з використанням нейронних мереж є актуальною задачею.

## II. МЕТА ДОСЛІДЖЕННЯ

Метою роботи є створення нейронної мережі, яка буде автоматично виявляти артефакти на КТ- зображеннях.

Розробка штучної нейронної мережі для виявлення і аналізу артефактів на КТ-зображеннях потребує створення нової математичної моделі, а також її програмної або апаратної реалізації, яка побудована за принципом організації та функціонування біологічної нейронної мережі, наприклад мережі нервових клітин живого організму. Навчання штучної нейронної мережі з математичного погляду передбачає вирішення багатопараметричної задачі нелінійної оптимізації [15].

Розроблена мережа вважається навченою, якщо функція втрат набуває мінімального значення.

## III. ПРОГРАМНІ ЗАСОБИ РОЗРОБКИ НЕЙРОННОЇ МЕРЕЖІ

Середовищем розробки програмного забезпечення та реалізації алгоритму нейронної мережі є Python та його бібліотеки: NumPy (Numerical Python), Tensor Flow, Keras API. Однією з ключових переваг програмування на мові Python є інтерпретаційний характер його коду. Інтерпретатор Python і його стандартні бібліотеки доступні у двійковій чи вихідній формі та можуть безперерійно працювати на всіх основних операційних системах, що забезпечує переносимість розробленого алгоритму на інші апаратні платформи.

Бібліотека NumPy дає змогу інтегрувати у розроблене програмне забезпечення математичні та логічні операції над масивами, перетворення Фур'є, операції, пов'язані з лінійною алгеброю. Бібліотека NumPy містить також колекцію інструментів і прийомів, за допомогою яких можна вирішити на комп'ютері математичні моделі для задач в галузі науки та техніки. Одним із таких програмних інструментів є багатовимірний об'єкт масиву, який є потужною структурою даних для ефективного обчислення масивів та матриць для задачі обчислення неронної мережі [30].

У бібліотеці TensorFlow представлені алгоритми машинного навчання, такі як обчислювальні графи та ребра, які дають змогу описати вершини чи вузли операції, а також представляють потоки даних між цими операціями [31].

Бібліотека Keras є основою для побудови глибоких нейронних мереж на Python. Keras дозволяє будувати як сучасні, глибокі системи навчання, а також ті, які використовуються в Google та Facebook, з невеликою складністю, а також з кількома рядками коду [33].

Для визначення якості роботи створеної нейронної мережі використано функцію втрат (loss function). За таку функцію обирали евклідову відстань, середньоквадратичну похибку або функцію кросентропії [17].

## IV. СТРУКТУРА НЕЙРОННОЇ МЕРЕЖІ ДЛЯ ВИЯВЛЕННЯ АРЕФАКТІВ

Сучасними програмними методами

оптимізації є AdaGrad (adaptive gradient algorithm), RMSProp (Root Mean Square Propagation), калманівський метод стохастичного градієнтного спуску (Kalman-based Stochastic Gradient Descent – KSGD) та метод Adam (Adaptive Moment Estimation) [20 – 22]. Ці методи є модифікаціями методу стохастичного градієнту і передбачають зміни коефіцієнта швидкості навчання під час навчання нейронної мережі.

В архітектурі розробленої нейронної мережі закладено апріорні знання з предметної області комп'ютерного зору: піксель зображення сильніше пов'язаний з сусіднім (локальна кореляція) і об'єкт на зображенні може зустрітися у будь-якій частині зображення [34].

Розроблена згорткова нейронна мережа являє собою чергування згортальних шарів (convolution layers), агрегувальних шарів (subsampling layers) і повнозв'язних шарів (fully-connected layer) на виході. Всі три види шарів можуть чергуватися в довільному порядку (рис. 1).



Рис. 1. Архітектура розробленої загорткової нейронної мережі.

У згортковому шарі мережі нейрони, які використовують однакові ваги, об'єднуються в карти ознак (feature maps), а кожен нейрон карти ознак пов'язаний з частиною нейронів попереднього шару. При обчисленні мережі враховується, що кожен нейрон виконує згортку деякої області попереднього шару, яка визначається групою нейронів, що пов'язані з даним нейроном [34].

У повнозв'язному шарі кожен нейрон з'єднаний з усіма нейронами мережі на попередньому рівні, причому кожен зв'язок має свій ваговий коефіцієнт.

Створена архітектура нейронної мережі реалізована з використанням бібліотеки для машинного навчання TensorFlow та нейромережевої бібліотеки Keras. Архітектура фінальної моделі нейронної мережі використовує нейронну мережу прямого

поширення, таку як багатшаровий перцептрон (multilayer perceptron) [21]. Кожен перцептрон у згортковому та повнозв'язному шарі має логічну схему з трьома виходами (рис.2).

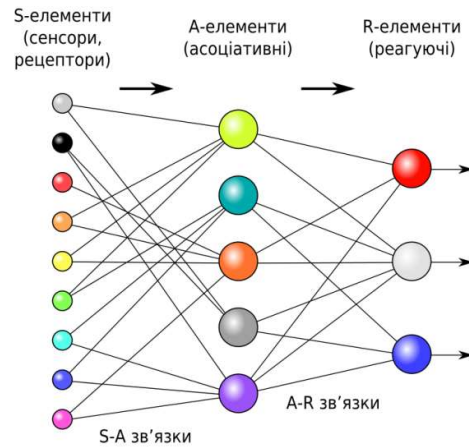


Рис. 2. Логічна схема перцептрон у шарі з трьома виходами.

Матриця ваг нейронної мережі має невеликий розмір (ядро згортки), вона «рухається» по всьому оброблюваному шару нейронів [34]. Проходячи від одного шару до наступного, приховані нейрони у шарі обчислюють зважену суму входів на них з урахуванням зваженої ваги в попередньому шарі, та застосовують нелінійну функцію активації до отриманого результату.

## V. ЗАСТОСУВАННЯ НЕЙРОННОЇ МЕРЕЖІ ДО КТ-ЗОБРАЖЕНЬ

Для формування тренувального та тестувальних наборів було відібрано КТ-зображення 10 різних людей з наявними металевими об'єктами в щелепно-лицевій області. Зображення були взяті з відкритої бази даних DeepLesion [12].

Після отримання зображень було створено тренувальний набір даних, який складався з 180 КТ-зображень в аксіальному виді (90 знімків з наявним артефактом та 90 знімків без артефакту). Для тестового та валідованих наборів було виділено по 20 зображень (10 знімків з наявним артефактом та 10 без артефакту).

На рисунку 3 представлено приклади КТ-зображень з артефактами, які було використано для навчання мережі.

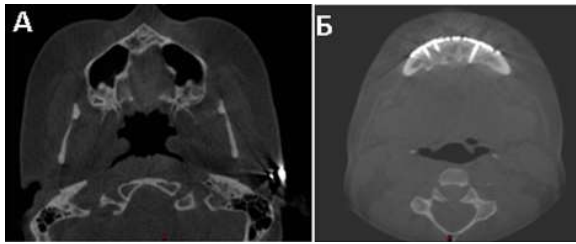


Рис. 3. КТ-зображення з наявними артефактами від металу: а) зображення з артефактом у вушному регіоні; б) зображення з металевими об'єктами в регіоні нижньої щелепи.

На рисунку 4 наведено приклади КТ-зображень, на яких відсутні будь-які металеві об'єкти.

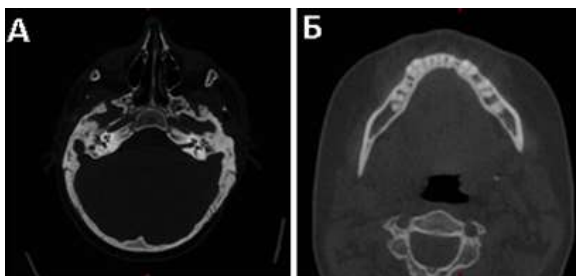


Рис. 4. КТ-зображення з відсутніми артефактами від металу: а) зображення регіону черепа; б) зображення нижньої щелепи.

На першому кроці розробки системи було створено базу згортку нейронної мережі з невеликою кількістю шарів – 6.

У подальшому архітектуру нейронної мережі було ускладнено для збільшення точності роботи системи шляхом збільшення кількості згортальних шарів у мережі до 9. Щоб обмежити межу ускладнення структури нейронної мережі прийнято гіпотезу, що модель архітектури мережі є адекватною, якщо результат розпізнавання наявних артефактів з металу на КТ-зображеннях має вірогідність більш ніж 95%, оскільки розроблена система буде використовуватись в медичних цілях.

Архітектуру моделі мережі з 9 згортальних шарів, яку було побудовано у Python 3 за допомогою бібліотеки Keras, наведено на рисунку 5.

Model: "sequential\_1"

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 200, 200, 32)	896
max_pooling2d_1 (MaxPooling2D)	(None, 100, 100, 32)	0
conv2d_2 (Conv2D)	(None, 100, 100, 64)	18496
max_pooling2d_2 (MaxPooling2D)	(None, 50, 50, 64)	0
conv2d_3 (Conv2D)	(None, 50, 50, 128)	73856
max_pooling2d_3 (MaxPooling2D)	(None, 25, 25, 128)	0
flatten_1 (Flatten)	(None, 80000)	0
dense_1 (Dense)	(None, 128)	10240128
dense_2 (Dense)	(None, 1)	129
Total params: 10,333,505		
Trainable params: 10,333,505		
Non-trainable params: 0		

Рис. 5. Скомпільована у Python 3 базова модель нейронної мережі

Тренування CNN моделі відбувалось на тестовій вибірці, яка складається із 180 зображень. На рисунку 6 наведено графік тренування моделі, а саме залежність функції втрат від номеру епохи. Кількість епох через які пройшла мережа було обрано 100, так як саме цього було достатньо для тренування мережі.

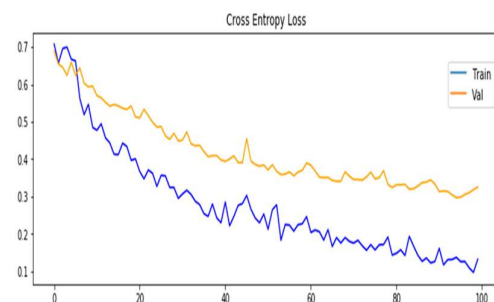


Рис. 6. Графік залежності функції втрат від номеру епохи для загорткової архітектури CNN

На рисунку 7 наведено графік залежності значення точності розпізнавання від номеру епохи.

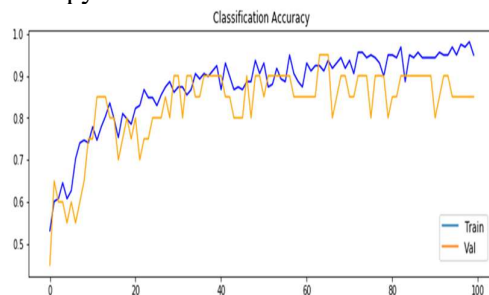


Рис. 7. Графік залежності точності від номеру епохи для загорткової архітектури CNN

З даних рисунків видно, що нейрона

мережа є стабільною та дієвою. Тренування пройшло успішно, проте максимальний отриманий результат точності складе 95%, що недостатньо для вирішення нашої задачі. Значення функції втрат даної становить 0.22.

Отриманий результат задовольняє вимогам, які були поставлені до системи, проте потребує порівняння з більш складними архітектурами нейронних мереж для того, щоб перевірити можливість досягнення більшої точності розпізнавання артефактів з металу.

## VI. ПОРІВНЯННЯ РЕЗУЛЬТАТІВ ОТРИМАНИХ НА ОСНОВІ ІНШИХ НЕЙРОННИХ МЕРЕЖ

У якості мережевої архітектури для порівняння результатів виявлення наявних артефактів розглянемо архітектуру AlexNet, яка застосовується для розпізнавання елементів біомедичних зображень.

Архітектура AlexNet складається з п'яти згорткових шарів, за деякими з яких слідує maximum pooling шар і потім три повністю пов'язані шари та, нарешті, 1000-ний класичний softmax класифікатор. Архітектуру даної моделі мережі, яку було побудовано у Python 3 за допомогою бібліотеки Keras, зображено на рисунку 8.

Layer	Feature Map	Size	Kernel Size	Stride	Activation	
Input	Image	1	227x227x3	-	-	
1	Convolution	96	55 x 55 x 96	11x11	4	relu
	Max Pooling	96	27 x 27 x 96	3x3	2	relu
2	Convolution	256	27 x 27 x 256	5x5	1	relu
	Max Pooling	256	13 x 13 x 256	3x3	2	relu
3	Convolution	384	13 x 13 x 384	3x3	1	relu
4	Convolution	384	13 x 13 x 384	3x3	1	relu
5	Convolution	256	13 x 13 x 256	3x3	1	relu
	Max Pooling	256	6 x 6 x 256	3x3	2	relu
6	FC	-	9216	-	-	relu
7	FC	-	4096	-	-	relu
8	FC	-	4096	-	-	relu
Output	FC	-	1000	-	-	Softmax

Рис. 8. Скомпільована у Python 3 модель AlexNet

Результати тренування даної моделі мережі з архітектурою AlexNet на нашій тренувальній вибірці показали прийнятні результати щодо залежності функції втрат від номеру епохи (рис. 9) та залежності точності від номеру епохи (рис. 10).

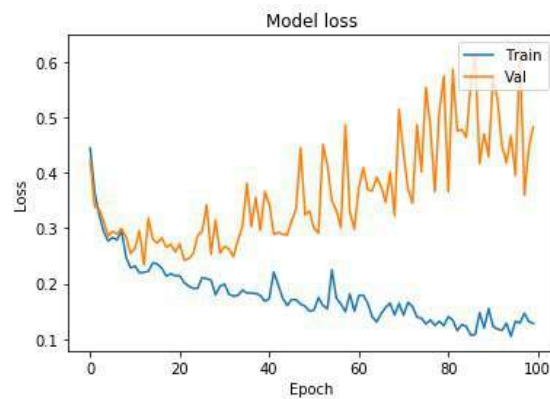


Рис. 9. Графік залежності функції втрат від номеру епохи для архітектури AlexNet

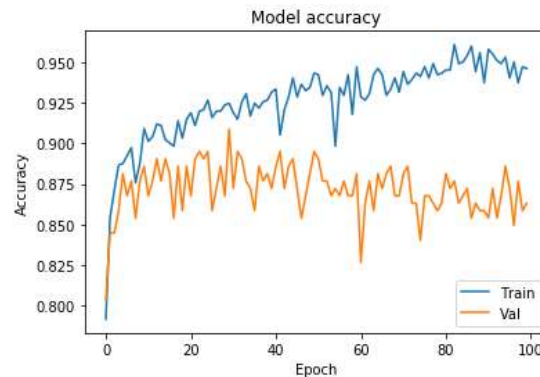


Рис. 10. Графік залежності точності від номеру епохи для архітектури AlexNet

Результати тренування моделі AlexNet показують, що дана модель починаючи з 20 епохи починає сильно перенавчатись. Хоча архітектура цієї моделі є більш складною ніж згорткова архітектура CNN моделі з 9 шарами.

Отже, архітектура AlexNet показує набагато гірший результат: максимальне значення точності – 87%, значення функції втрат – 0.6.

Наступна архітектура нейронної мережі для порівняння результатів, яка використовується для виявлення артефактів на біомедичних зображеннях – VGG16.

Архітектура VGG16 складається з дванадцяти згорткових шарів, за деякими з яких слідує maximum pooling шари і потім чотири повністю пов'язані шари та, нарешті, 1000-ний класичний softmax класифікатор. Архітектуру даної моделі мережі, яку було побудовано у Python 3 за допомогою бібліотеки Keras, зображено на рисунку 11.

Layer	Feature Map	Size	Kernel Size	Stride	Activation	
Input	Image	1	224 x 224 x 3	-	-	
1	2 X Convolution	64	224 x 224 x 64	3x3	1	relu
	Max Pooling	64	112 x 112 x 64	3x3	2	relu
3	2 X Convolution	128	112 x 112 x 128	3x3	1	relu
	Max Pooling	128	56 x 56 x 128	3x3	2	relu
5	2 X Convolution	256	56 x 56 x 256	3x3	1	relu
	Max Pooling	256	28 x 28 x 256	3x3	2	relu
7	3 X Convolution	512	28 x 28 x 512	3x3	1	relu
	Max Pooling	512	14 x 14 x 512	3x3	2	relu
10	3 X Convolution	512	14 x 14 x 512	3x3	1	relu
	Max Pooling	512	7 x 7 x 512	3x3	2	relu
13	FC	-	25088	-	-	relu
14	FC	-	4096	-	-	relu
15	FC	-	4096	-	-	relu
Output	FC	-	1000	-	-	Softmax

Рис. 11. Скопійована у Python 3 модель мережі VGG16

Результати тренування даної моделі мережі з архітектурою VGG16 на нашій тренувальній вибірці показали дещо кращі результати щодо залежності функції втрат від номеру епохи (рис. 12).

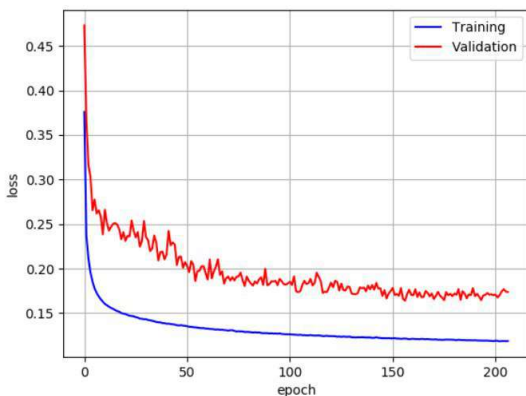


Рис. 12. Графік залежності функції втрат від номеру епохи для архітектури VGG16

Результати тренування моделі VGG16 показують, що дана модель також перенавчається, проте не так значно як модель AlexNet. Максимальне значення точності для даної моделі – 90%, значення функції втрат – 0.2.

Отже, архітектура VGG16 показує прийнятний результат відносно залежності точності від номеру епохи, проте поступається згортковій архітектурі CNN моделі з 9 шарами.

## VII. ОБГОВОРЕННЯ РЕЗУЛЬТАТІВ

У роботі створенно та перевірено на тестовій вибірці нейронні мережі, яка дають змогу автоматично виявляти наявні артефакти

на КТ- зображеннях, зокрема із включенням металу.

Виходячи з тренування та порівняння моделей нейронних мереж (CNN, AlexNet, VGG16) складено таблицю 1, яка дає змогу наочного представити отримані результати.

Таблиця 1

### Показники точності мережевих моделей

Модель	Точність на тренувальному наборі	Точність на валідаційному наборі	Точність на тестовому наборі
CNN (6 шарів)	85%	80%	83%
CNN (9 шарів)	100%	94%	95%
AlexNet	89%	86%	87%
VGG16	88%	92%	90%

Найкращі результати щодо виявлення артефактів на КТ-зображеннях показує 9 шарова CNN модель. Більш складні моделі (AlexNet, VGG16) не показали достатньо вирогідних результатів, що показує непридатність цих моделей для поставленої задачі. Збільшення епох навчання для таких моделей призводить до їх перенавчання, що і впливає на остаточний результат щодо точності розпізнавання.

Отже, в якості фінальної моделі для задачі виявлення наявних артефактів на КТ-зображеннях доцільно використання 9 шарової CNN моделі, яка показала найкращі результати точності – 95 %.

## ПЕРЕЛІК ПОСИЛАНЬ

- [1] Цифровые технологии в современной реконструктивной хирургии / [О. С. Карпов, С. С. Гаврюшин, М. Н. Замятин та ін.]. // Вестник Национального медико-хирургического Центра им. Н.И. Пирогова. – 2016.
- [2] Кокорев П. А. АНАЛИЗ АРТЕФАКТОВ ИЗОБРАЖЕНИЙ В КОМПЬЮТЕРНОЙ ТОМОГРАФИИ / П. А. Кокорев. // Научно-технический вестник информационных технологий, механики и оптики 2008 ВАК. – 2008. – С. 84–87.
- [3] Календер В. Компьютерная томография – М.: Техносфера, 2006. – 244 с.
- [4] Марусина М.Я., Казначеева А.О. Современные виды томографии. Учебное пособие. – СПб: СПбГУ ИТМО, 2006. – 132 с.
- [5] Гордина Г.С. Методика проведения и оценки результатов компьютерной томографии пациентов с аномалиями зубочелюстной системы до хирургического лечения [Электронный ресурс] / Гордина Г.С.. – 2016. – Режим доступа до ресурсу: <https://vrach-aspirant.ru/articles/stomatology/13443/>.

- [6] Лавренюк М. С. Огляд методів машинного навчання для класифікації великих обсягів супутникових даних / М. С. Лавренюк, О. М. Новіков. // *System Research & Information Technologies*. – 2018. – С. 52–71.
- [7] Bishop C.M. *Pattern Recognition and Machine Learning* / C.M. Bishop. — NY: Springer. — 2006.
- [8] Zgurovsky M.Z. *System Analysis: Theory and Applications* / M.Z. Zgurovsky, N.D. Pankratova // Springer. — 2007. — 448 p.
- [9] Novikov A. The synthesis of information protection systems with optimal properties / A. Novikov, A. Rodionov // *Complexity and Security*. — Vol. 37. — 2008. — 307 p.
- [10] LeCun Y. Deep learning / Y. LeCun, B. Yoshua, H. Geoffrey // *Nature*. — Vol. 521, N 7553. — 2015. — P. 436–444.
- [11] Ghamisi P. Advanced Spectral Classifiers for Hyperspectral Images: A review / P. Ghamisi, J. Plaza, Y. Chen et al // *IEEE Geoscience and Remote Sensing Magazine*. — Vol. 5, N 1. — 2017. — P. 8–32.
- [12] Fukunaga K. *Introduction to statistical pattern recognition* / K. Fukunaga // Academic press, 2013. — 591 p.
- [13] Maulik U. Remote Sensing Image Classification: A survey of support-vector-machine-based advanced techniques / U. Maulik, D. Chakraborty // *IEEE Geoscience and Remote Sensing Magazine*. — Vol. 5, N 1. — 2017. — P. 33–52.
- [14] Gislason P.O. Random forests for land cover classification / P.O. Gislason, J.A. Benediktsson, J.R. Sveinsson // *Pattern Recognition Letters*. — Vol. 27, N 4. — 2006. — P. 294–300.
- [15] McCulloch W.S. A logical calculus of the ideas immanent in nervous activity / W.S. McCulloch, W. Pitts // *The bulletin of mathematical biophysics*. — Vol. 5, N 4. — 1943. — P. 115–133.
- [16] Rumelhart D.E. Learning representations by back-propagating errors / D.E. Rumelhart, G.E. Hinton, R.J. Williams // *Cognitive modeling*. — Vol. 5, N 3. — 1988. — P. 213–220.
- [17] Amari S.I. Statistical theory of learning curves under entropic loss criterion / S.I. Amari, N. Murata // *Neural Computation*. — Vol. 5, N 1. — 1993. — P. 140–153.
- [18] Schmidhuber J. Deep learning in neural networks: An overview / J. Schmidhuber // *Neural networks*. — Vol. 61. — 2015. — P. 85–117.
- [19] Rumelhart D.E. Learning internal representations by error propagation / D.E. Rumelhart, G.E. Hinton, R.J. Williams // *Parallel distributed processing*. MIT Press. — 1986. — P. 318–362.
- [20] Duchi J. Adaptive subgradient methods for online learning and stochastic optimization / J. Duchi, E. Hazan, Y. Singer // *Journal of Machine Learning Research*. — 2011. — P. 2121–2159.
- [21] Patel V. Kalman-Based Stochastic Gradient Method with Stop Condition and Insensitivity to Conditioning / V. Patel // *SIAM Journal on Optimization*. — Vol. 26, N 4. — 2016. — P. 2620–2648.
- [22] Kingma D. Adam: A method for stochastic optimization / D. Kingma, J. Ba // *arXiv preprint arXiv*. — Vol. 1412, N 6980. — 2014. — P. 1–15.
- [23] Haykin S. *Neural networks and learning machines* / S. Haykin // Upper Saddle River. — NJ, USA: Pearson, 2009. — Vol. 3. — 938 p.
- [24] Hornik K. Multilayer feedforward networks are universal approximators / K. Hornik, M. Stinchcombe, H. White // *Neural networks*. — Vol. 2, N 5. — 1989. — P. 359–366.
- [25] Liu P. SVM or deep learning? A comparative study on remote sensing image classification / P. Liu, K.K.R. Choo, L. Wang, F. Huang // *Soft Computing*. — Vol. 21, N 23. — 2017. — P. 7053–7065.
- [26] Huang F.J. Large-scale learning with SVM and convolutional nets for generic object categorization / F.J. Huang, Y. LeCun // *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. — 2006. — P. 284–291. DOI: 10.1109/CVPR.2006.164.
- [27] Pirotti F. Benchmark of machine learning methods for classification of a Sentinel-2 image / F. Pirotti, F. Sunar, M. Piragnolo // *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*. — Vol. 41. — 2016. — P. 335–340.
- [28] Breiman L. Random forests / L. Breiman // *Machine learning*. — Vol. 45, N 1. — 2001. — P. 5–32.
- [29] Програмування числових методів мовою Python : підруч. / А. В. Анісімов, А. Ю. Дорошенко, С. Д. Погорілий, Я. Ю. Дорогий ; за ред. А. В. Анісімова. — К. : Видавничо-поліграфічний центр "Київський університет", 2014. — 640 с.
- [30] Методичні вказівки до виконання лабораторних робіт з дисципліни “Інженерія прикладних інтелектуально-орієнтованих програмних продуктів” для студентів спеціальностей 121 “Інженерія програмного забезпечення” та 122 “Комп’ютерні науки та інформаційні технології” (всіх форм навчання) / В.М. Льовкін. — Запоріжжя : ЗНТУ, 2016. — 80 с.
- [31] Goldsborough P. A Tour of TensorFlow / P. Goldsborough. // *Fakultät für Informatik*. — 2016.
- [32] Understand TensorFlow by mimicking its API from scratch [Електронний ресурс] // *Machine Learning*. — 2019. — Режим доступу до ресурсу: <https://medium.com/@d3lm/understand-tensorflow-by-mimicking-its-api-from-scratch-faa5578170d>.
- [33] My Journey into DeepLearning using Keras [Електронний ресурс] // *Medium*. — 2018. — Режим доступу до ресурсу: <https://towardsdatascience.com/my-journey-into-deeplearning-using-keras-part-1-67cbb50f65e6>.
- [3] Карпович А. В. Використання загорткових нейронних мереж для задачі класифікації текстів / А. В. Карпович. // *International scientific journal «Internauka»*. — 2018. — С. 69–72.

УДК 616.1–616.7

# НЕЙРОННАЯ СЕТЬ ДЛЯ ВЫЯВЛЕНИЯ АРТЕФАКТОВ НА КТ-ИЗОБРАЖЕНИЯХ

*Шлыков В.В.*, доц., д.т.н.

[v.shlykov@kpi.ua](mailto:v.shlykov@kpi.ua)

*Волянник О.М.*, магистр

[volyanik.oleg@gmail.com](mailto:volyanik.oleg@gmail.com)

Кафедра биомедицинской инженерии

Национальный технический университет Украины

«Киевский политехнический институт имени Игоря Сикорского»

г. Киев, Украина

**Реферат** – Обоснована необходимость создания системы для выявления артефактов на КТ-изображениях, создана архитектура нейронной сети в программной среде Python с помощью нейросетевой библиотеки Keras и проверена ее точность с помощью тестового набора изображений. Показано, что потенциально возможности полученной системы могут быть усовершенствованы для других задач биомедицинской инженерии.

**Ключові слова** – компьютерная томография, нейронная сеть, свертка нейронной сети, артефакты, Python, библиотека Keras, 3D-шаблон протеза.

UDC 616.1–616.7

# NEURAL NETWORK FOR DETECTION OF ARTIFACTS ON CT-IMAGES

*Shlykov V.V.*, Associate Professor, Ph.D.

[v.shlykov@kpi.ua](mailto:v.shlykov@kpi.ua)

*Volyanik O.M.*, Master student

[volyanik.oleg@gmail.com](mailto:volyanik.oleg@gmail.com)

Department of Biomedical Engineering

National Technical University of Ukraine

“Igor Sikorsky Kyiv Polytechnic Institute”

Kiev, Ukraine

**Abstract** – The necessity of creating a system for detecting artifacts on CT images was substantiated, the neural network architecture was created in Python software using the Keras neural network library, and its accuracy was tested using a test set of images. It is shown that, potentially, the capabilities of the resulting system can be improved for other biomedical engineering tasks.

**Keywords** – CT, neural network, convolutional neural network, artifacts, Python, Keras library, 3D prosthesis template.